UNIVERSITY OF SOUTHAMPTON

Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

An Investigation into Dense Material Segmentation

by

Yuwen Heng

ORCiD: 0000-0003-3793-4811

Supervisor: Hansung Kim
Second Supervisor: Srinandan Dasmahapatra
A thesis for the degree of
Doctor of Philosophy

September 2023

University of Southampton

Abstract

Faculty of Engineering and Physical Sciences
School of Electronics and Computer Science

Doctor of Philosophy

An Investigation into Dense Material Segmentation

by Yuwen Heng

The dense material segmentation task aims at recognising the material for every pixel in daily images. It is beneficial to applications such as robot manipulation and spatial audio synthesis. However, achieving accurate material segmentation for 3-channel RGB images is challenging due to the considerable variation in the appearance of a material. This research aims to design high-performance material segmentation networks that can achieve an accuracy above 80% and serve real-time inference. In this thesis, three and a half contributions will be introduced and analysed to accomplish the research objective.

The proposed networks extend the idea of combining material and contextual features for material segmentation. Material features describing transparency and texture can generalise to unseen images regardless of material appearances such as shape and colour. Contextual features can reduce the segmentation uncertainty by providing extra global or semi-global information about the image, such as the scene and object categories.

Contribution A investigates the possibility to leverage contextual features without extra labels. In particular, the boundaries between different materials are selected as semi-global contextual information. A self-training approach is adopted to fill in the unlabelled pixels in the sparsely labelled datasets, and a hybrid network named Context-Aware Material Segmentation Network (CAM-SegNet) is introduced to extract and combine the boundary and material features.

Contribution B.1 explores the way to extract material features from cross-resolution image patches which takes the variation in pixel area covered by each material into account. The Dynamic Backward Attention Transformer (DBAT) is proposed to explicitly gather the intermediate features extracted from cross-resolution patches and merge them dynamically with predicted attention masks.

Contribution B.2 studies the features that networks learn to make predictions. By

analysing the cross-resolution features and the attention weights, this study interprets how the DBAT learns from image patches. The features are further aligned to semantic labels by performing network dissection, which emphasises that the proposed model can extract material-related features better than other methods.

Contribution C proposes to segment materials with recovered hyperspectral images which theoretically offer distinct information for material identification, as variations in the intensity of electromagnetic radiation reflected by a surface depend on the material composition of a scene. The proposed Material Hyperspectral Network (MatSpectNet) leverages the principles of colour perception in modern cameras to regularise the reconstructed hyperspectral images and employs the domain adaptation method to generalise the hyperspectral reconstruction capability from a spectral recovery dataset to material segmentation datasets. The reconstructed hyperspectral images are further filtered using learned response curves and enhanced with human perception (such as roughness) to learn reliable material features.

The proposed networks are evaluated quantitatively and qualitatively using two open-access material segmentation datasets. CAM-SegNet demonstrates strong discriminative ability when trained with material boundaries, enabling it to accurately identify materials with similar appearances. With cross-resolution patch features, DBAT can accurately segment materials with varying shapes. It has also been demonstrated to extract material-related features more proficiently than other networks. The MatSpecNet, embraced with the recovered hyperspectral images, yields the best performance (88.24% in the averaged per-pixel accuracy), and excels at identifying the material under different illumination conditions, particularly with the presence of spotlight reflection.

Contents

List of Figures

List of Tables

Declaration of Authorship

I declare that this thesis and the work presented in it is my own and has been generated by me as the result of my own original research.

I confirm that:

1. This work was done wholly or mainly while in candidature for a research degree at this University;
2. Where any part of this thesis has previously been submitted for a degree or any other qualification at this University or any other institution, this has been clearly stated;
3. Where I have consulted the published work of others, this is always clearly attributed;
4. Where I have quoted from the work of others, the source is always given. With the exception of such quotations, this thesis is entirely my own work;
5. I have acknowledged all main sources of help;
6. Where the thesis is based on work done by myself jointly with others, I have made clear exactly what was done by others and what I have contributed myself;
7. Parts of this work have been published as:
   Publications as the first author:
   - Yuwen Heng, Srinandan Dasmahapatra, and Hansung Kim. Material recognition for immersive interactions in virtual/augmented reality. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 577–578, 2023.
   - Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Enhancing material features using dynamic backward attention on cross-resolution patches. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022a

- Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Cam-segnet: A context-aware dense material segmentation network for sparsely labelled datasets. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 190–201. INSTICC, SciTePress, 2022b. ISBN 978-989-758-555-5.

Publications as a coauthor:

- Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation for a single omnidirectional image with reversed-gradient warming-up thresholds discriminator. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023
- Mona Alawadh, Yihong Wu, Yuwen Heng, Luca Remaggi, Mahesan Niranjan, and Hansung Kim. Room acoustic properties estimation from a single 360° photo. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 857–861, 2022.
- Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation from a single omnidirectional image using domain adaptation. In *European Conference on Visual Media Production (CVMP)*, pages 1–9, 2021

Signed:..... Date:.....

Acknowledgements

I would like to express my deep gratitude and appreciation to my supervisors, Dr Hansung Kim and Dr Srinandan Dasmahapatra, for their invaluable guidance and support throughout my research. Their vast knowledge, critical insights, and unwavering dedication have been instrumental in shaping my research achievements. I am grateful for the opportunities they have provided me to develop my skills, challenge myself, and learn from their expertise.

I am also indebted to Jiawen Chen and other faculty members of Baidu for providing me with their GPUs to train my networks. Their support was critical to the successful completion of this research. Their generosity allowed me to experiment with a range of deep learning models and optimise them for the best performance. I also appreciate their willingness to provide me with feedback and guidance throughout the coding process. Their contributions have been invaluable to my research, and I am honoured to have had the opportunity to work with such a talented and dedicated team.

I would also like to express my heartfelt gratitude to my colleagues and friends, especially Yihong Wu and Mona, for their unwavering support and encouragement throughout my research. Yihong Wu's insightful discussions and feedback on my ideas were invaluable, and Mona's validation of my design with downstream tasks, such as immersive sound rendering, was critical to the success of my work. In addition, I am grateful to my other colleagues and friends who have shared their expertise, advice, and experiences with me. Their camaraderie, motivation, and intellectual discussions have been instrumental in my academic and personal development.

Lastly, I would like to express my sincere gratitude to my family, who have always been my pillars of strength, motivation, and inspiration. Their love, support, and encouragement have sustained me throughout my academic journey. I am thankful for their sacrifices, understanding, and unwavering belief in me. I want to explicitly thank my wife, Danni Li, who stayed with me for three years, wherever I went. Her unwavering love, patience, and encouragement have been my driving force in pursuing this degree. I am also grateful to my parents, Weiguo Heng and Ruixiao Ying, for their love, encouragement, and financial support. Without them, I would not be where I am today.

Definitions and Abbreviations

CAM-SegNet	Context-Aware Material Segmentation Network
BCAM-SegNet	Boundary CAM-SegNet
DBA	Dynamic Backward Attention
DBAT	Dynamic Backward Attention Transformer
MatSpectNet	Material Hyperspectral Network
LMD	Local Material Dataset
MINC	Material in Context Dataset
DLMD	dense LMD
MCubeS	MultiModal Material Segmentation Dataset
NIR	Near Infra-Red
PBR	Physically Based Rendering
CKA	Centered Kernel Alignment
SOTA	State-Of-The-Art
Pixel Acc	Averaged Per-Pixel Accuracy
Mean Acc	Mean Class Accuracy
mIoU	Mean Intersection over Union
FoV	Field-of-View
VISAPP	International Conference on Computer Vision Theory and Applications
CCIS	Communications in Computer and Information Science series
AAAI	Advancement of Artificial Intelligence
BMVC	British Machine Vision Conference
VRW	IEEE Conference on Virtual Reality and 3D User Interfaces
FC	Fully Connected Layer
FCN	Fully Convolutional Network
Conv	Convolutional Layer
CNNs	Convolutional Neural Networks
NLP	Natural Language Processing
SAM	Segment Anything Model
U-Net	U-shaped network
CRF	Conditional Random Field
RGFS	Region-Guided Filter Selection
BiSeNet	Bilateral Segmentation Network

ViT	Vision Transformer
FPN	Feature Pyramid Network
CRFasRNN	Conditional Random Fields as Recurrent Neural Networks
Conv-CRF	Convolutional CRF
PAC-CRF	Pixel-Adaptive Convolutional CRF
MLP	Multi-Layer Perceptron
$HSIC_1$	Unbiased Estimator of the Hilbert-Schmidt Independence Criterion
ToF	Time-of-Flight
TPSF	Temporal Point Spread Functions
BRDF	Bidirectional Reflectance Distribution Function
d_w	Camera Working Distance
$S(x)$	Spectral Recovery Network
$R(h)$	sRGB Transformation Network
y_{gt}	Ground Truth Segments
y_{pd}	Predicted Segments
y^b	Boundary Map
c	Category c
C	Number of Available Categories
P	Precision
R	Recall
BF_1	Boundary Metric
$L_{boundary}$	Boundary Loss
L_{focal}	Focal Loss
Q	Query
K	Key
V	Value
H	Height
W	Width
K, L	Two Gram matrices
$\tilde{K}, \tilde{L}$	Metrics with Zero Diagonal Entries
a, b	Two Vectors
m	Number of Samples
tr	Function to Sum the Matrix Diagonal Elements
a_k	99.5% Value Threshold
$V(\lambda)$	Photopic Reflectance
$M(\lambda)$	Melanopic Reflectance
O_L	Local Branch Output
O_G	Global Branch Output
O_C	Composite Branch Output
X_G	Global Branch Feature Map
X_L	Local Branch Feature Map

X_{CG}	Concatenated Global Branch Feature Map
X_{CL}	Concatenated Local Branch Feature Map
S_t	Student Model at Round t
p.p.	Percentage Points
$x, \mathcal{X}$	RGB Images
$h, \mathcal{H}$	Hyperspectral Images
$S, S(x)$	Spectral Recovery Network
$R, R(h)$	RGB Transformation Network
μ	Brightness Factor
ν	Possion Noise Level
α	Learning Rate
σ	Gaussian Noise Level
$P(\nu)$	Possion Distribution
$N(0, \sigma)$	Gaussian Distribution
W_{rgb}	RGB Response Functions
f_{noise}	System Noise
rgb_{noisy}	Noisy RGB Images
rgb_{clean}	Noiseless RGB Images
f_{jpeg}	JPEG Transformation
$\Delta band_i$	Sensitivity Difference at Frequency i
S	Spectra Shape Matrix
x_m	Ground Truth Image from LMD
$\hat{x}_m$	Recovered LMD Image by $R(h)$
x_s	Ground Truth Image from ARAD_1K
$\hat{x}_s$	Recovered ARAD_1K Image by $R(h)$
$\odot$	Element-wise Multiplication
$\oplus$	Element-wise Summation

Chapter 1

Introduction

The dense material segmentation task aims to recognise the physical material categories (*e.g.* metal, plastic, stone, etc.) for each pixel in the input image. The material cues can provide critical information to many applications, such as robot manipulation (Zhao et al., 2020a; Shrivatsav et al., 2019) and spatial audio synthesis (McDonagh et al., 2018; Kim et al., 2019; Chen et al., 2020a). One example is to teach a robot to perform actions such as 'cut' with a tool. This action indicates that the robot should grasp a knife at the wooden grip and cut with the metal blade (Shrivatsav et al., 2019). For scenarios that can harm the human body, *e.g.*nuclear garbage collection, robots need material labels to put the waste into corresponding bins (Zhao et al., 2017a). Materials can also be used to estimate the acoustic properties (how sound interacts with surroundings (Delany and Bazley, 1970)) from physical material categories to synthesise immersive sound with spatial audio reflections and reverberation (McDonagh et al., 2018; Kim et al., 2019; Tang et al., 2020). Moreover, physically based rendering (PBR) (Hodaň et al., 2019), which is the technique used to synthesise realistic camera or Light Detection and Ranging (LiDAR) outputs, requires the material optical properties such as surface reflectivity in many applications, for example, autonomous driving simulation (Eversberg and Lambrecht, 2021).

One of the main challenges in the dense material segmentation task is that materials could have a variety of appearances, including colour, shape, and transparency (Fleming, 2014). The appearances vary when viewed in different contexts, such as objects and places (Schwartz and Nishino, 2020). For example, a metal knife is glossy under bright lighting conditions, but a rusted metal mirror can be dull. In order to achieve high accuracy, an ideal network should know all possible combinations; thus, a large dataset is necessary. However, the similarity between the appearances of different materials can make annotation work challenging. Even humans cannot identify a material precisely from rectangular RGB images, especially when the material is covered with a coat of paint (Bell et al., 2013a). Consequently, material datasets are often

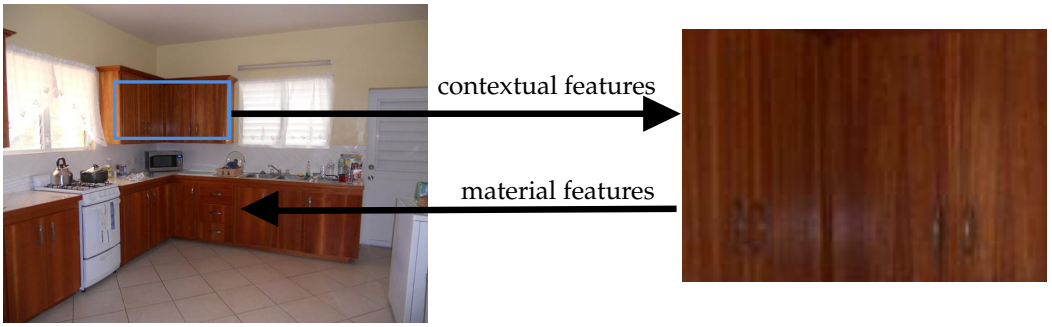


FIGURE 1.1: The kitchen image with a wooden cupboard.

sparsely labelled regarding the number of images and the integrity of labelled material regions. For example, the training set segments in the Local Material Database (LMD) (Schwartz and Nishino, 2016, 2020) cover only a tiny region of the material, as shown in the ground truth images in Figure 3.4. The ground truth segments in OpenSurfaces are visualised and analysed in Appendix A.1.

As suggested by Schwartz and Nishino (2020, 2016), a possible solution is to combine material and contextual features (Schwartz, 2018; Schwartz and Nishino, 2020, 2016; Bell et al., 2015b; Heng et al., 2022b). Material features allow the network to identify the categories despite their varied appearances, and contextual features can limit the possible categories of materials that appear in a given scene. Taking Figure 1.1 as an example, given that the picture is taken in a kitchen, then the cupboard is probably made of wood. Here the kitchen and cupboard provide the contextual information that semantically describes the scene of the image and the object to which the material belongs. When zooming in on the cupboard, the tree-grain pattern, which is a unique texture of wood that humans can name, describes the appearance of the material. In turn, the generalisable features of wood can be used to infer labels for areas of the kitchen image that are also made of wood.

Unfortunately, combining material and contextual features remains a challenging task. Schwartz and Nishino (2020, 2016) proposed a multi-branch network architecture (Zhang et al., 2020b, 2019b). The network adopts one branch to extract material features from image patches, and multiple pre-trained branches targeting object segmentation and scene recognition tasks to extract contextual features. The material and contextual features are concatenated to predict the material labels. They hold the belief that training a network with small image patches cropped from material regions without contextual cues can force the network to learn from the visual properties of materials. Although their work contributed a feasible method to achieve dense material segmentation with a neural network, the network design is still immature. First, the pre-trained branches that provide contextual features are not fine-tuned together with the material branch since dedicated material datasets do not contain contextual labels. Second, the

patch resolution is fixed for all images in a dataset, which does not consider the varying areas that materials cover within and across images. Finally, it should be noted that learning material features from image patches is an implicit method that is heavily reliant on the training process and cannot guarantee that the resulting features are directly related to the material properties.

To elevate the performance of dense material segmentation to an acceptable level (with an accuracy exceeding 80%), this thesis addresses the aforementioned three challenges by proposing the design of innovative network architectures that incorporate both material and contextual features. The ensuing sections expound on the research contributions.

1.1 Proposed Methodologies

1.1.1 First PhD stage

During the initial nine months of my PhD research, the challenge of segmenting materials is explored using contextual features without the need for extra labels and pre-trained networks. The networks pre-trained on object recognition or scene classification are not used in this research since they tend to perform badly on unseen data due to the misalignment of training data distribution (Song et al., 2019), and the learned features are not controllable and may not contribute to the material segmentation task well. Examining the network training process, it becomes evident that there are a lot of randomnesses that can affect the features learned by a network across multiple runs. For example, the randomly initialised trainable parameters vary the starting point of the training. Moreover, factors such as the optimiser, the learning rate scheduler, and the data feeding order affect the gradient decay path. The network can be converged to different positions at the parameter space. Consequently, the features that these networks learn to make predictions can vary. Moreover, Raghu et al. (2021) quantitatively show that networks with different architectures can learn different features with the Centered Kernel Alignment (CKA) matrix (Nguyen et al., 2020; Raghu et al., 2021). Since it is impossible to fully control the features that a network learns, the contextual features extracted from pre-trained networks may not be suitable for material segmentation.

Instead of integrating fully untangled material and contextual features, this report demonstrates that a carefully designed mechanism to combine these features during training can improve the segmentation performance. As the first step towards a high-performance material segmentation network, Chapter 3 proposes the CAM-SegNet, which shares the intermediate features across branches during training. The proposed CAM-SegNet consists of global, local and composite branches. The global branch is

responsible for extracting contextual features from the entire image, while the local branch is designed to learn the material features from image patches. Finally, the composite branch produces the final predictions from merged features. Chapter 3 demonstrates the efficiency of CAM-SegNet by adjusting the global branch to extract boundary-related contextual features with the loss function that measures the alignment between predicted and ground truth material boundaries (Bokhovkin and Burnaev, 2019).

In order to leverage the boundary features for material segmentation, it is essential that the material regions are densely labelled. However, existing datasets such as LMD are sparsely labelled where the categories of pixels near the material boundaries are unknown. Therefore, in section 3.3.3, a self-training approach is adopted to annotate the unknown pixels with predicted labels. To evaluate the network's performance, in addition to the test set, eight more images from the LMD test set are manually labelled and referred to as the dense LMD (DLMD). The proposed CAM-SegNet achieves an improvement of 3-20% in averaged Per-Pixel Accuracy (Pixel Acc) and 6-28% in Mean Intersection over Union (mIoU) compared to recently published network architectures and single-branch approaches in the control group. Moreover, the experiments also indicate that the BCAM-SegNet ensures that accuracy does not decline with the iterative self-training approach.

The contributions of the first stage were published in the *17th International Conference on Computer Vision Theory and Applications* (VISAPP) at the end of month nine. These contributions were further extended and included in the Springer book *Computer Vision, Imaging and Computer Graphics Theory and Applications* during the summer, which is part of the Communications in Computer and Information Science series (CCIS).

1.1.2 Second PhD stage

Having shown that sharing the features across branches during training can improve network performance, in my second PhD stage between month nine and month eighteen, I managed to answer the research question of how to segment materials with multiple patch resolutions efficiently. When extracting material features, existing networks (Schwartz and Nishino, 2020, 2016, 2013) choose to use a fixed patch resolution, which may not be the best choice for all images. As affected by the camera working distance d_w and field-of-view (FoV), the areas that materials cover vary within and across images. Ideally, small patch resolution should be applied to the boundary region to separate mutually enmeshed materials, and large patch resolution can be used to cover as much information as possible for the region belonging to a single material. Moreover, since multi-branch networks can introduce inevitable overheads leading to low frame rates as well as unacceptable training times, the networks should be simplified for real-time inference.

Instead of searching for a fixed patch resolution, a simple yet effective single-branch transformer architecture called DBAT is devised in Chapter 4 to aggregate cross-resolution patch features. Inspired by the hierarchical architecture of Swin transformer (Liu et al., 2021b), which gradually merges image patches to get a global view, Section 4.3.1 proposes a Dynamic Backward Attention (DBA) module to aggregate the intermediate features extracted from image patches with different resolutions. Concretely, a transformer feature map from a shallow layer contains features extracted from local patches (Raghu et al., 2021), especially when using window-based self-attention. The proposed DBAT aggregates multiple intermediate feature maps to identify the materials with cross-resolution patch features since the patch resolution varies with the depth of the layer. To cope with the flexibility of d_w and FoV, a set of pixel-wise attention masks, which represent the dependency on each patch resolution, are applied in the DBA module to dynamically aggregate the feature maps. These masks are calculated from the deepest feature map (Map_4 in Figure 4.2) since it holds a relatively global perspective of the input image. Before feeding the aggregated feature into the decoder, Section 4.3.2 further proposes a feature merging module which ensures the aggregated feature can learn complementary features through an attention-based residual connection.

The effectiveness of the proposed DBAT is examined through a comparison with recently published segmentation networks that can achieve real-time performance (at least 24 frames per second). The DBAT reaches a median Pixel Acc of 86.85% on the LMD, which is 21.21% higher than the CAM-SegNet trained in Chapter 3 and outperforms the second-best model evaluated in Chapter 4 by 2.15%. The DBAT was published at the *33rd British Machine Vision Conference* (BMVC).

However, similar to other network-based methods, the DBAT faces challenges in terms of interpretability. Ascertaining whether the network genuinely acquires material features through numerical evaluation or segmentation visualisation is a complex task. Therefore, during the summer of my second PhD stage, I further endeavour to interpret the network behaviour of the DBAT in Chapter 5, using statistical and visual tools such as calculating the attention equivalent patch size, visualising attention masks, and assessing the CKA heatmap (Nguyen et al., 2020; Raghu et al., 2021). In order to interpret the features with human-readable concepts, the network dissection method (Zhou et al., 2018; Bau et al., 2017, 2019, 2020) is also employed to identify the features learned by the network by aligning layer neurons with semantic concepts. By analysing the semantic concepts of the extracted features, Chapter 5 illustrates that the DBAT excels in extracting material-related features, such as texture, which is an essential property for distinguishing between various materials. By comparing the semantic concepts of features extracted by other networks trained with either material or object datasets, the results also indicate that the network architecture can influence the extracted features, and the patch-based design is indeed effective in compelling the networks to segment images based on material features.

The contributions of DBAT, along with its interpretability, were submitted to the *IEEE Transactions on Image Processing* in my third PhD stage, and the journal paper is currently under review. The application of DBAT in immersive sound rendering has been accepted by the *30th IEEE Conference on Virtual Reality and 3D User Interfaces* (VRW) as a poster.

1.1.3 Third PhD stage

During months eighteen to twenty-seven of my third PhD stage, my research focus shifted from investigating material features in RGB images to hyperspectral images. The research question addressed during this stage was how to explicitly learn reliable material features. Although recent studies and the outcomes from my first two PhD research stages show that it is possible to achieve acceptable performance with annotated RGB datasets (Heng et al., 2022b,a; Schwartz, 2018; Schwartz and Nishino, 2020; Bell et al., 2015a), the networks are learning material features implicitly. In the meantime, the experiment in (Liang et al., 2022; Mao et al., 2022) shows that additional measurements of light such as near infra-red (NIR) and laser beam reflection can distinguish materials more robustly. The theory is that the spectral profile of reflected electromagnetic waves, which is an explicit measurement of material property, is unique to various materials (Saragadam and Sankaranarayanan, 2020; Lichtman and Conchello, 2005; Colthup et al., 1990). Since spectral cameras (Behmann et al., 2018) can capture the spectral profile of surface materials, it is feasible to use the hyperspectral images they produce for material segmentation.

While hyperspectral imaging has been widely used in geoscience and remote sensing (Zhong et al., 2016; Kalman and Bassett III, 1997; Li et al., 2022b; Xue et al., 2021; Mehta et al., 2021; Liu et al., 2019a) over twenty years, the cost of collecting hyperspectral images hinders its widespread adoption in material segmentation for daily scenes (Stuart et al., 2022). A spectral camera can take a long acquisition time to scan a megapixel hyperspectral image with sufficient signal-noise ratio since the same amount of light has to be sampled at hundreds of wavelength bands (Behmann et al., 2018; Zhang et al., 2019a). This problem necessitates concessions in image spatial and spectral resolution. In addition, the ambient light should be able to cover the entire operating spectrum range, so the spectral camera should be used under daylight or halogen-based illumination. Before taking the hyperspectral images, the camera has to be calibrated with the measurement of black and white reference samples to analyse the material reliably (Behmann et al., 2018; Shaikh et al., 2021). The stringent lighting requirements further restrict the application of hyperspectral images in indoor and motion scenes.

In order to make spectral information more accessible for computer vision applications, researchers have been working on recovering spectral information from more easily obtainable data sources, such as RGB images (Arad et al., 2022). Over the past three years,

several methods (Li et al., 2020; Hu et al., 2022; Cai et al., 2022c) have successfully improved the quality of reconstructed hyperspectral images. However, it remains unclear how these methods generalise to images captured by different camera models, as this aspect has not been explicitly investigated. In consideration of this problem, Chapter 6 proposes a novel MatSpectNet to enhance the quality of recovered hyperspectral images on material datasets lacking RGB-hyperspectral image pairs. Figure 6.1 shows that the proposed MatSpectNet consists of two main sections. The network first learns to recover the hyperspectral images with the physical model of the camera, which serves as a constraint to ensure that the hyperspectral images preserve their physical property. Then the recovered hyperspectral images are processed with a multi-layer perceptron (MLP) to learn the material features from the spectral information at each pixel.

To understand the proposed approach, it would be useful to delve into image theory first. An image is the quantitative measurement of the radiation from an illumination source or reflected by scene elements. Similar to the human perception system, the RGB camera measures the radiance of the visible spectrum with red, green and blue spectral response functions that accumulate the electromagnetic radiation from 380 to 720 nanometers and produce the raw-RGB values for each pixel (Magnusson et al., 2020). The raw-RGB image is further processed with in-camera non-linear transformations including brightness adjustment and gamma correction to produce the final sRGB image, which is the format used in image datasets.

The proposed MatSpectNet model incorporates the physical relationship between hyperspectral and RGB images based on the image theory to regularise the colour metamerism, which states the many-to-one mappings from hyperspectral to RGB images. Specifically, Chapter 6 exploits the fact that recovered hyperspectral images can be transformed into the original RGB counterparts through known spectral response functions and in-camera processing. This physical constraint is a key feature of the model and enables it to make reliable material predictions based on the recovered hyperspectral images.

However, for open-access material segmentation datasets such as LMD (Schwartz and Nishino, 2020) and OpenSurfaces (Bell et al., 2013a), the spectral response functions and in-camera image-processing pipeline are unknown. To bring hyperspectral images to the material segmentation task, the MatSpectNet models the physical camera with the sRGB transformation network $R(h)$ that contains trainable components to adjust the unknown parameters of the camera model. As illustrated in Figure 6.1, given an sRGB image x, the MatSpectNet optimises that $R(S(x)) = x$ where $S(x)$ is the spectral recovery network that recovers hyperspectral images from sRGB ones. In practice, $S(x)$ is pre-trained on the ARAD_1K dataset (Arad et al., 2022) and fine-tuned together with material datasets. To align the data distribution, the idea of domain adaptation is used during training (Wu et al., 2021).

The recovered hyperspectral images are further processed with learned spectral response filters followed by a MLP to extract per-pixel material features. The spectral response filter is similar to the RGB spectral response functions in the mechanism, which aggregates the spectral information based on the sensitivity to the spectrum at each wavelength. The per-pixel material features are then tagged with the surface properties such as specularity and roughness queried from the most similar spectral measurement in the spectraldb dataset (Jakubiec, 2022), which serves as a piece of additional evidence to identify the materials.

The proposed network, MatSpectNet, outperforms existing models in the material segmentation task. With a Pixel Acc of 88.24% and a Mean Acc of 83.82%, the network shows an improvement of 1.60%/3.42% over the DBAT. Notably, MatSpectNet is particularly adept at recognising material categories that have limited samples, even under varying light conditions such as spotlight reflection. These results are supported by per-category performance metrics and visualised segmentation results.

The contributions of MatSpectNet have been submitted to *2024 Association for the Advancement of Artificial Intelligence (AAAI) Conference on Artificial Intelligence* and the conference paper is still under review by the submission of this thesis.

1.2 Overview of Thesis Structure

The next chapter, Chapter 2 presents a comprehensive background study on the datasets, methods and techniques employed in the thesis, including material segmentation datasets, deep learning architectures, optimisation algorithms, evaluation metrics and network interpretability methods. The chapter also discusses the challenges and limitations of existing approaches in material segmentation in each subsections, and highlights the gaps in the current literature that the thesis aims to address. Additionally, Chapter 2 introduces some methods including PBR that are potentially useful but not utilised in this research. The characteristics of these methods are discussed and they can be saved for future research.

Chapter 3 introduces the first proposed network design, the CAM-SegNet, which combines contextual and material features to improve segmentation accuracy. Chapter 4 presents the second proposed network design, the DBAT, which leverages cross-resolution patch features to improve segmentation performance. Chapter 5 discusses the interpretability of the DBAT, and how it can aid in material analysis and understanding. Chapter 6 presents MatSpectNet, a novel architecture designed for material segmentation that incorporates the material optical properties from recovered hyperspectral images.

These chapters provide detailed descriptions of the network architectures, the training procedures, and the evaluation results on various datasets. These chapters also compare the performance of proposed networks with state-of-the-art (SOTA) methods in the literature, and discuss the advantages and limitations of the proposed approach.

Finally, Chapter 7 concludes this thesis with a comprehensive summary of the main contributions and findings. The proposed methods have demonstrated their effectiveness in the material segmentation task. Specifically, CAM-SegNet combines contextual and material features for accurate material segmentation without extra labels or pre-trained networks. DBAT enables cross-resolution feature extraction from image patches, and its interpretability supports network design intuition. Lastly, MatSpecNet is capable of estimating material categories by explicitly recovering material optical properties. Moreover, Chapter 7 discusses the limitations and future directions of the proposed methods, and suggests potential research directions in material segmentation.

1.3 Contributions

The main contributions are summarised in the following subsections for each year.

1.3.1 First Year: CAM-SegNet

- *Hybrid Network Architecture*. Instead of relying on pre-trained networks to provide contextual features, the proposed hybrid network CAM-SegNet combines extracted boundary features with material features so that no extra labels are needed and the contextual features can be trained together with material features.
- *Self-training Approach*. To provide boundary features for the CAM-SegNet, the sparsely labelled material segmentation dataset is annotated with predicted labels, following the procedure of the self-training approach.

1.3.2 Second Year: DBAT

- *Dynamic Backward Attention Transformer*. An effective module enhances material features by dynamically adjusting the dependency on cross-resolution patch features.
- *Feature Merging Module*. This module is composed of the attention mechanism and residual connection to guide the aggregation of cross-resolution patch features.

- *Network interpretability*. The network behaviour of DBAT is illustrated through a batch of methods including descriptive statistics as well as visualised images. In addition, the learned features are analysed with semantic labels to support the design intuition through network dissection (Bau et al., 2020).

1.3.3 Third Year: MatSpecNet

- *Physically-Constrained Spectral Recovery.* Based on the theory that sRGB values can be obtained from hyperspectral images with known spectral response functions and in-camera processing, the spectral recovery network is regulated with a trainable sRGB transformation.
- *Domain-Aware Network Training.* To leverage the spectral recovery dataset for material segmentation, the domain adaptation is used to alleviate data distribution discrepancy between spectral recovery and material datasets. Moreover, domain-specific spectral response functions and image-processing pipelines are constructed.
- *Interpretable Hyperspectral Processing.* The learned spectral filters aggregate the spectra across the bandwidth and infer the electromagnetic frequency that contributes to material segmentation most.
- *Multi-Modal Fused Material Segmentation.* The filtered per-pixel spectra and queried surface properties are fused together to make the material prediction from both spectral measurements as well as other empirical observations such as surface roughness (Jain et al., 2013; Jakubiec, 2016; Jones and Reinhart, 2017; Lucas et al., 2014).

1.4 Publications

The contributions that are published or under review at peer-reviewed conferences and journals are listed in the following subsections.

1.4.1 Accepted as First Author

1. Yuwen Heng, Srinandan Dasmahapatra, and Hansung Kim. Material recognition for immersive interactions in virtual/augmented reality. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 577–578, 2023.

2. Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Enhancing material features using dynamic backward attention on cross-resolution patches. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022a

3. Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Camsegnet: A context-aware dense material segmentation network for sparsely labelled datasets. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 190–201. INSTICC, SciTePress, 2022b. ISBN 978-989-758-555-5.

1.4.2 Accepted as Coauthor

1. Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation for a single omnidirectional image with reversed-gradient warming-up thresholds discriminator. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023

2. Mona Alawadh, Yihong Wu, Yuwen Heng, Luca Remaggi, Mahesan Niranjan, and Hansung Kim. Room acoustic properties estimation from a single 360° photo. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 857–861, 2022.

3. Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation from a single omnidirectional image using domain adaptation. In *European Conference on Visual Media Production (CVMP)*, pages 1–9, 2021

1.4.3 Under Review

1. *IEEE Transactions on Image Processing*: Chapter 4 and Chapter 5 are merged together as a journal paper which has been submitted for review.

2. *AAAI Conference on Artificial Intelligence*: Chapter 6 has been submitted to AAAI 2024 for review.

3. *Computer Vision, Imaging and Computer Graphics Theory and Applications*: Chapter 3 has been extended and submitted to CCIS book series for review. It has been approved in the preliminary analysis and forwarded to the chairs for final decision.

Chapter 2

Literature Review

This chapter starts with an overview of material datasets and their characteristics. Then, popular deep learning based segmentation networks for material segmentation are introduced. Furthermore, methods for interpreting and analysing the behaviour of these networks are discussed. In addition, this chapter covers the essential knowledge of material features and hyperspectral images, which are crucial for developing reliable material segmentation models. Finally, data synthesis methods for generating synthetic training data to enhance network performance are briefly reviewed. Overall, this chapter provides a comprehensive understanding of the techniques in material segmentation and their associated methodologies.

2.1 Material Segmentation Datasets

In the computer vision domain, the material can be considered as an additional property that provides valuable clues for identifying an object. Therefore, researchers have chosen to add material segments to existing object datasets. For instance, Farhadi et al. (2009) enhanced the Pascal dataset[1] (Everingham et al., 2005), Zheng et al. (2014) extended the NYU dataset[2] (Silberman et al., 2012), both with eight material attributes. These material attributes include both material categories and material traits. For example, the attributed Pascal dataset contains four material categories, such as 'metal' and 'plastic', and four material traits, such as 'furry' and 'shiny'. Although their work initiated the research on material segmentation, the limited number of material categories is insufficient for applications that require dense material segmentation, such as room acoustic rendering (Kim et al., 2019).

[1] https://vision.cs.uiuc.edu/attributes/
[2] https://kylezheng.org/research-projects/densesegattobj/

In order to meet the requirements of a dedicated material segmentation dataset, Bell et al. (2013b) created OpenSurfaces[3]. This is the first large-scale, high-resolution material dataset, which contains more than one hundred thousand segments belonging to 45 categories. However, the dataset is highly unbalanced, with some categories, such as 'sponge', containing only a few samples. In subsequent work, Bell et al. (2015b) extended OpenSurfaces with additional image samples in the unbalanced categories and then organised the dataset into 23 mutually exclusive material categories. This new dataset is named Material in Context (MINC[4]). However, the training set of MINC only contains labelled pixels instead of segments, making it non-trivial to train a neural network end-to-end.

Recently, Schwartz and Nishino (2020) released the LMD[5], which contains 5,845 images with material segments that each cover a single category. This dataset is further extended with material traits such as soft and fuzzy (Schwartz and Nishino, 2019). The authors carefully chose 16 material categories without controversial ones such as 'carpet' and 'sky' in OpenSurfaces and MINC. The well-annotated segments and dedicated material categories make LMD the most suitable material dataset for effective deep learning methods. The drawback is that this dataset is coarsely labelled. First of all, the number of samples is insufficient since LMD is very diverse in terms of material categories and scenes. Second, the ground truth segments may not cover all pixels belonging to the same category, as shown in Figure 3.4. Accordingly, it is difficult for networks to recognise the materials precisely, especially for pixels near the boundary. Despite its drawbacks, LMD remains the most suitable dataset for material segmentation tasks, particularly for indoor applications. As such, the experiments in this thesis focus on evaluating the proposed networks using LMD as the preferred dataset.

In the year 2022, Liang et al. (2022) released the MultiModal Material Segmentation dataset (MCubeS[6]), which captures the visual appearance of materials in daily outdoor scenes from the viewpoint of self-driving scenarios. This dataset uses three different imaging modalities: RGB, polarization, and NIR, and it contains 500 sets of multimodal images capturing 42 street scenes. The images are densely annotated based on 20 distinct categories, including ground truth material segmentation categories such as plastic and fabric, and semantic segmentation categories such as leaf and human body. Cai et al. (2022a) annotated the well-established KITTI dataset densely with 20 material categories and published the KITTI-Materials dataset[7]. This dataset contains 1,000 images collected from 24 driving scenes. Although the MCubeS dataset provides a valuable resource for researchers to develop and evaluate multimodal material segmentation methods for outdoor scenes, the KITTI-Materials dataset can be combined

[3]http://opensurfaces.cs.cornell.edu/
[4]http://opensurfaces.cs.cornell.edu/publications/minc/
[5]https://vision.ist.i.kyoto-u.ac.jp/codeanddata/localmatdb/
[6]https://vision.ist.i.kyoto-u.ac.jp/research/mcubes/
[7]https://vision.ist.i.kyoto-u.ac.jp/research/rgbrms/

with KITTI to provide both material and object labels, my research focuses on indoor applications, especially immersive sound rendering, and therefore these two datasets are not evaluated in this thesis.

2.2 Introduction to Neural Networks for Segmentation Tasks

Before stepping into the material segmentation task, this section introduces the fundamental concepts of neural networks and the networks dedicated to segmentation tasks. In particular, the Fully Convolutional Network (FCN) (Long et al., 2015) will be introduced. FCN was originally proposed as a network architecture for segmentation tasks, capable of handling images of any resolution. It has since been widely adopted in recent advancements (Mo et al., 2022) and extended to various segmentation architectures. The importance of this network topology is particularly evident when dealing with datasets such as LMD, which contains images gathered from various sources with a wide range of resolutions. By providing an overview of these concepts, this section will provide a solid foundation for understanding the material segmentation networks.

2.2.1 Foundations of Neural Networks

From the perspective of mathematics, a basic neural network such as AlexNet (Krizhevsky et al., 2017) can be considered as a series of matrix multiplication (one matrix is the input image or last layer feature map, the other one is the network kernel) followed by non-linear transformations (Liu et al., 2019b) such as the rectified linear unit (Agarap, 2018). The mathematical operation that takes a set of inputs, performs a computation on them, and produces an output is called a neuron in neural networks. It is modelled after the structure and function of a biological neuron in the brain (Lin, 2017). These operations are applied to the input data in a hierarchical manner, constructing a complex non-linear mapping $f_\theta(x) = \hat{y}$ that links input x to its predicted label $\hat{y}$. During training, the network parameters θ are adjusted in a certain way to minimize the average loss $\bar{l}(y, \hat{y})$. The loss function measures the difference between $\hat{y}$ and its corresponding ground-truth label y for all training samples in a dataset. The convolutional (Conv) kernel is the most important network kernel widely used to process images and achieve dense segmentation. The Conv kernel operates by convolving (the operation is the sum of the Hadamard product, also known as the element-wise product, represented as $\odot$ in this thesis) a small filter over the input image, producing a set of feature maps that capture local patterns and correlations between neighbouring pixels. The size, stride, and padding of the Conv kernel can be tuned to control the receptive field of the network and its spatial resolution. Convolutional neural networks (CNNs) leverage the power of Conv kernels to extract hierarchical representations of visual features, allowing them

to learn complex and abstract image representations that are critical for various image analysis tasks, including recognition, detection, and segmentation.

2.2.2 Network Architectures for Segmentation Tasks

Networks designed for image classification tasks downsample and flatten the resolution of feature maps to obtain a reasonable label (Krizhevsky et al., 2017; Huang et al., 2017; He et al., 2016). These networks are referred to as encoders since they produce a highly compressed and abstract description of the image. However, networks designed for segmentation tasks need to produce labels for every pixel in the input image, whereas encoders can only generate a single label description for the entire input image. Moreover, the network kernel used to predict the label is the Fully Connected (FC) kernel (Basha et al., 2020), where each neuron is connected to every neuron in the previous layer. As a consequence, the network can only work with a pre-defined image resolution. In consideration of this problem, Long et al. (2015) proposed the first FCN architecture trained end-to-end to achieve dense segmentation. They first train a classification network to identify the category of the central pixel of an image patch, then replace the FC kernel with a Conv kernel of size 1×1. This allows the network to work with the original images instead of patches and generate an output mask accordingly. In the material segmentation realm, Bell et al. (2015a), who proposed the material segmentation dataset MINC, adopts the idea of FCN as an early attempt to solve the dense material segmentation task.

The output mask of FCN is then upsampled to generate a segmentation mask that matches the resolution of the input image (Badrinarayanan et al., 2017). This process is known as a decoder in segmentation tasks, which is necessary to ensure that the segmentation predictions are precise and aligned with the input image spatial coordinates. The upsampling can be done using different methods, such as transposed convolution (Im et al., 2019), bilinear interpolation (Smith, 1981), or nearest-neighbour interpolation (Han, 2013). The choice of method can affect the quality of the segmentation map and the computational cost of the network. Therefore, selecting the appropriate upsampling method is a crucial factor in designing efficient and accurate segmentation networks.

Ronneberger et al. (2015) further improved this architecture to a more elegant U-shaped network (U-Net), which contains a downsampling encoder to extract features, and an upsampling decoder to recover the shape and make predictions. The U-Net also contains skip-connections that copy and paste extracted features from the encoder, which may ease the optimisation problem (Liu et al., 2020). This U-Net architecture won the 2015 ISBI neuronal structures segmentation challenge. Since then, the encoder-encoder FCN architecture and its successors tend to dominate 2D and 3D segmentation challenges, such as the Cityscapes task (Ghiasi and Fowlkes, 2016; Chen et al., 2018; Tao

et al., 2020) and the SemanticKITTI benchmark (Milioto et al., 2019; Zhu et al., 2020). They show that networks following the FCN design topology can make dense segmentation predictions with high accuracy as well as good segment boundaries. The proposed networks in this thesis also follow this encoder-decoder FCN architecture.

2.2.3 Material Segmentation Networks

Recent achievements using datasets mentioned in Section 2.1 are also based on the FCN architecture. Since every dataset has its flaws, extra processes are necessary to segment images densely. For MINC that contains only pixel label (Bell et al., 2015b), a classifier is trained to recognise the central point of the image patch, which covers about 5% [8] of the area of the whole image. After that, the FC kernel is replaced with an equivalent Conv kernel (Long et al., 2015) and the global pooling layer is removed to work as a sliding-window segmentation network. Finally, the nearest-neighbour interpolation operation upsamples the predicted segments to the same size as the input images. Although their best attempt achieved an accuracy of 79.8%, more advanced segmentation network structures with trainable decoders are not suitable for the MINC since its training set contains no labelled segments.

For LMD, since the training set contains single-material segments, it is possible to achieve dense material segmentation with an end-to-end segmentation network. Schwartz and Nishino (2013) claimed that for the material segmentation task, it is better to train a network with cropped image patches (without contextual cues about object and scene) to force the network to focus on the generalisable material features. Schwartz and Nishino (2016) then discovered that integrating contextual information can reduce the uncertainty in identifying materials. They proposed a network which takes 48×48 image patches as input and concatenates contextual features before the final layer. The contextual features are extracted from two parallel network branches, pre-trained on the ADE20K[9] (Zhou et al., 2017) and the SUN[10] (Xiao et al., 2010, 2016) separately. While their method showed improvement in segmentation performance, it has the drawback of running three branches simultaneously, which incurs unacceptable computing resources, especially for real-time applications. Additionally, since the contextual branches are not fine-tuned with LMD, they may not be able to extract high-quality contextual features. Furthermore, the fixed patch resolution does not account for the varying material area, limiting its effectiveness. This thesis aims to address the aforementioned limitations by proposing two novel networks: CAM-SegNet and DBAT. CAM-SegNet learns contextual and material features coherently during training, and DBAT learns material features from cross-resolution image patches.

[8] The patch size is 23.3% of the smaller image dimension and can cover up to 5.29% of the area of the image

[9] https://groups.csail.mit.edu/vision/datasets/ADE20K/

[10] https://vision.princeton.edu/projects/2010/SUN/

Although the patch training method is designed to extract material features, what features networks learn to achieve high accuracy remains a mystery. Recently, Geirhos et al. (2019) proved that neural networks tend to depend on texture instead of shape in object recognition tasks when trained with full-size images. Since the texture is a vital material feature, it is possible to design a network that captures both material and contextual features without a dedicated material branch. Zhao et al. (2017a, 2020a) proposed a network to learn from MINC test set, which contains about two thousand segmented images. Their experiments indicate that a single-branch network can work well for material segmentation if the dataset has well-annotated segments. However, the MINC test set they used is insufficient to give reliable conclusions. Therefore, the proposed methods in this thesis are evaluated with LMD and Opensurfaces since these two datasets have more high-quality segments than MINC test set. In particular, the single-branch network DBAT introduced in Chapter 4 utilises transfer training (Torrey and Shavlik, 2010) to incorporate contextual features. The encoder is first trained with the object classification task, and then transferred to the material segmentation task with a randomly initialised decoder.

The presented publications have provided sufficient foundational knowledge to commence the research study in material segmentation. Moreover, limitations and challenges of the published material segmentation models have been identified, such as the computational cost of parallel branches and fixed patch resolution. These insights will provide valuable guidance for the research study to address these limitations and propose innovative solutions. Some of the training strategies discussed above utilised the Conditional Random Field (CRF) to refine the segmentation result to get a clear boundary between different materials. This thesis will introduce CRF further in Section 2.3.1.

While conducting the research, it is important to consider and take into account the concurrent contributions made by other research teams such as MCubeSNet (Liang et al., 2022) and RMSNet (Cai et al., 2022a). Similar to the research in this thesis, MCubeSNet also focused on establishing new network architectures to improve segmentation accuracy, though they are interested in integrating different imaging modalities by designing a new decoder and this research is intended to investigate new encoders to learn material-related features from RGB images. The Region-Guided Filter Selection (RGFS) is a key component of the MCubeSNet decoder, which learns and selects the features learned from different modalities based on a guidance field to produce the final material segmentation so that different materials integrate those imaging modalities in a way most relevant to identify them correctly. The guidance field is a semantic segmentation mask that identifies the region of each object, generated by training a segmentation network using RGB images and annotated masks from CityScapes (Cordts et al., 2016). The concept is based on the premise that the occurrence of materials and object

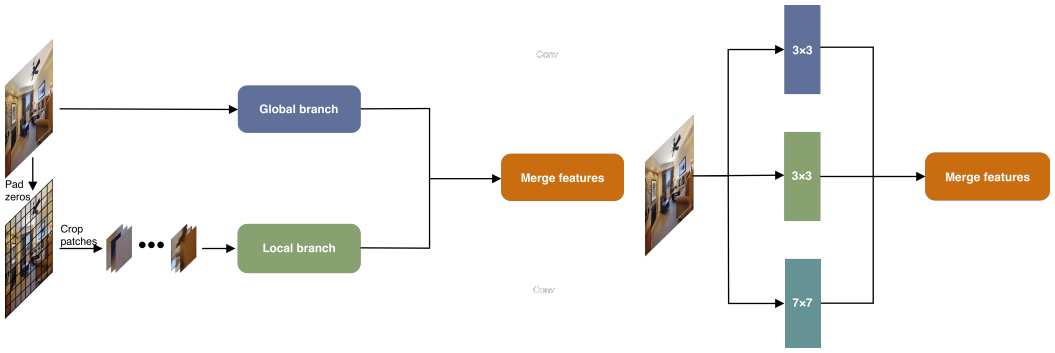
instances are strongly correlated, which also supports the design of combining contextual and material features. Regarding the RMSNet (Cai et al., 2022a), it is important to note that this work is an independent and concurrent effort in comparison to my DBAT approach. Both methodologies share a common goal of integrating multi-level features with transformers for material segmentation. However, there are distinct differences in the specific objectives of each approach. The RMSNet approach aims to combine material and contextual cues, leveraging their complementary nature to enhance material segmentation performance. In contrast, the primary focus of DBAT, my proposed method, lies in aggregating cross-resolution patch features. This approach enables the effective integration of information from patches of varying resolutions, facilitating accurate and comprehensive material segmentation with varying material areas.

2.2.4 Global and Local Networks

Global and local network architecture provides an approach to combine features extracted from full-size images by the global branch and image patches by the local branch. Chen et al. (2019) adopted this approach to preserve local details when processing downsampled images. Due to the memory bottleneck when processing high-resolution patches, they split these patches into multiple batches and gather the full feature maps with several forward steps. This method makes the feature combining process complicated and costs more training time. To reduce the training time, Zhang et al. (2020b) reduced trainable parameters by sharing the weights between local and global branches. Wu et al. (2020) alleviated the training burden by proposing only critical patches to refine the global segmentation. Likewise, Iodice and Mikolajczyk (2020) proposed to crop the extracted global feature maps into equal blocks as the local features. As for the dense material segmentation task, the CAM-SegNet adopts this architecture to compensate for the lost features when training with a single branch alone. According to Schwartz (2018), the network trained with original images tends to ignore material features, while the network trained with patches drops contextual cues. Moreover, LMD contains no high-resolution images so that the CAM-SegNet can jointly train the global and local branches in an end-to-end manner without a severe training burden.

2.2.5 Multiscale Networks

The multiscale network is a neural network architecture designed to process an input image at multiple scales. It typically consists of several parallel kernels or network branches, each of which extracts features from the input image at a different scale. For instance, some popular multiscale networks include the Pyramid Scene Parsing Network (PSPNet) (Zhao et al., 2017b), which uses pooling layers of multiple kernel sizes



(a) Global and local network Architecture (b) Multiscale network Architecture

FIGURE 2.1: The network architecture comparison between global-local network and multiscale network.

to produce feature maps of varying scales; DeepLabV3+ (Chen et al., 2018), which employs dilated Conv kernels of different dilation spacing to increase the receptive field; and the Bilateral Segmentation Network (BiSeNetV2) (Yu et al., 2021a), which comprises two pathways with different downsample rates. The output of each kernel or branch is combined to generate a final prediction. Multiscale networks are commonly used in computer vision tasks as they allow the network to capture both fine-grained details and global context information.

Although multiscale networks may sound similar to global and local networks, they are in fact different types of network architectures. As shown in Figure 2.1, one key difference is that global and local networks are designed to combine features learned from both image patches and the full-size image, whereas multiscale networks are intended to learn features at different scales from the full-size image. Multiscale networks can be used as the global branch in a global and local network. In fact, the proposed methods in this thesis choose the Feature Pyramid Network (FPN) (Lin et al., 2017a) as the decoder to gradually upsample the resolution with the intermediate multiscale features extracted by the encoder through lateral connections.

2.3 Boundary Refinement

For the dense material segmentation task, the network-based methods may not predict the pixels near the boundary accurately due to the lack of training labels to refine the boundary quality (Schwartz and Nishino, 2016). One possible solution is to use the CRF appended to the output of the segmentation network (Section 2.2) to refine the segmentation quality. Another possible way to refine the boundary is to use the boundary loss (Bokhovkin and Burnaev, 2019), which measures the proportion of overlapping boundary pixels between ground truth and predicted segments. This section will provide a detailed introduction to the theory of these two methods.

2.3.1 Conditional Random Fields

CRF is a powerful tool to predict labels with the knowledge of neighbouring pixels (Sutton and McCallum, 2006). For the image segmentation task, CRF optimises two penalties: the single pixel prediction should be the same as ground truth label (also known as the unary term), and the assumption that adjacent pixels should have the same class label (the pairwise term, as shown in Equation 2.1). Here $\hat{y}$ is the predicted label for a pixel, w_p is the weight of this pairwise term in the loss function, δ is the Potts label compatibility function. $\delta = 1$ if $\hat{y}_i \neq \hat{y}_j$ else 0. k is the unit Gaussian kernel, which measures the difference between $\mathbf{f}_i$ and $\mathbf{f}_j$, where $\mathbf{f}$ is decided by the pixel position and raw pixel value in the original image.

$$\psi_{ij}(\hat{y}_i, \hat{y}_j) = w_p \delta(\hat{y}_i \neq \hat{y}_j) k(\mathbf{f}_i - \mathbf{f}_j) \tag{2.1}$$

The research of CRF focuses on how to decide that two pixels are neighbours and should be classified as the same category. Krähenbühl and Koltun (2013) proposed the well-known dense-CRF, which assumes that a pixel is adjacent to all other pixels. Although dense-CRF is powerful for material segmentation (Bell et al., 2015b), the parameters cannot be optimised together with the network. Moreover, tuning the parameters manually can be a time-consuming task. According to the supplemental code in (Bell et al., 2015b), the CRF refined predictions are sensitive to the parameter choices. To cope with the problem, Zheng et al. (2015) implemented the dense-CRF model as a recurrent neural network (CRFasRNN) so that the CRF parameters can be optimised together with the network. However, it is difficult to accelerate the training process of the CRFasRNN with GPU (Teichmann and Cipolla, 2019).

To speed up the training, this thesis evaluates two GPU-trainable CRF variants, the Convolutional CRF (Conv-CRF) (Teichmann and Cipolla, 2019) and the Pixel-adaptive Convolutional CRF (PAC-CRF) (Su et al., 2019). The Teichmann and Cipolla (2019) managed to implement a GPU trainable Conv-CRF, with the locality assumption that the pairwise term (Equation 2.1) is zero if the same Conv kernel does not cover the two pixels at the same time. They proved that the Conv-CRF segmentation performance is still comparable with the dense-CRF with the local assumption. At the same time, Su et al. (2019) proposed another GPU-trainable PAC-CRF. The PAC-CRF obeys the locality assumption and considers long-range dependency with the dilated kernel. Although PAC-CRF can predict segments more accurately compared with Conv-CRF (Su et al., 2019), it consumes more memory and requires longer computing time. For CAMSegNet, the evaluation of both of these two CRF methods is analysed in Section 3.5.3. The experiment demonstrates that the PAC-CRF method is susceptible to the impact of material texture, leading to the imposition of distinct categories on adjacent pixels

around the texture, and ultimately resulting in inaccurate predictions for material segmentation.

2.3.2 Boundary Loss Function

Another method to refine the boundary is to use a loss function that measures the quality of the segmentation boundary. One straightforward choice is the IoU between ground truth segments and predicted segments for each material category. Another way is to measure the overlapping between boundary pixels for each category. However, both methods are count-based, which is not differentiable, and thus cannot be used to train networks directly. Rahman and Wang (2016) proposed a Soft IoU loss, which adopts the continuous predicted output from the sigmoid layer in the IoU function. Bokhovkin and Burnaev (2019) utilised the max pooling operation to generate the segment boundaries for both ground truth segments as well as predicted segments after the sigmoid layer. The boundary loss is then computed based on a distance map that measures the distance from each pixel at the predicted segment boundary to the nearest pixel at the ground truth segment boundary and vice versa. To avoid the non-differentiable operation argmax, these two losses are defined directly on the score maps after the softmax operation. This thesis adopts the boundary loss in (Bokhovkin and Burnaev, 2019) to train the networks since it is explicitly designed to refine the boundaries.

For each material category c, the network would predict a probabilistic score map y^c_{pd} whose values are within the range [0,1] after the softmax operation. The values of the corresponding ground truth segments y^c_{gt} are in the set 0,1. In order to compute the alignment between y^c_{pd} and y^c_{gt}, the boundary map of category $y^{b,c}$ is defined with the following equation:

$$y^{b,c} = pool(1 - y^c, \theta_1) - (1 - y^c) \tag{2.2}$$

where θ_1 is the max pooling size. In order to calculate the Euclidean distances from pixels to boundaries, it is necessary to obtain a supporting map, which is an extension of the boundary map $y^{b,c,ext}$:

$$y^{b,c,ext} = pool(y^{b,c}, \theta_2) \tag{2.3}$$

Then the precision and recall P^c, R^c can be computed, by counting the points within the maximum distance defined by θ_2:

$$P^c = \frac{\sum(y^{b,c}_{pd} \odot y^{b,c,ext}_{gt})}{\sum(y^{b,c}_{pd})}, \; R^c = \frac{\sum(y^{b,c}_{gt} \odot y^{b,c,ext}_{pd})}{\sum(y^{b,c}_{gt})} \tag{2.4}$$

The boundary metric (BF_1^c) proposed by (Csurka et al., 2013) and the actual boundary loss $L_{boundary}$ (Bokhovkin and Burnaev, 2019) is then computed as:

$$BF_1^c = \frac{2P^cR^c}{P^c + R^c}, \; L_{boundary} = \frac{1}{C}\sum_c (1 - BF_1^c) \tag{2.5}$$

where C is the number of material categories. Although experiments in (Kang et al., 2021; Bokhovkin and Burnaev, 2019) have shown that this boundary loss can help the network to optimise the predictions near the boundaries, the loss value may not decrease when used in isolation since it does not contribute to the segmentation accuracy directly. Therefore, the local branch features, designed to achieve high accuracy, are passed to the global branch to ensure the BCAM-SegNet can extract boundary features steadily. Moreover, the boundary loss function assumes that the ground truth segments should cover all adjacent pixels belonging to the same category. As a consequence, the boundary loss cannot be used for the MINC and LMD databases directly. Section 2.4 introduces a solution, the self-training strategy, which can provide pseudo labels for the unlabelled pixels.

While conducting the research, Borse et al. (2021) proposed a new loss function which assumes that the boundaries of ground truth segments and the predicted segments are related to each other through a homography transformation (Liu et al., 2018a). The authors construct an inverse-transformation network that takes the boundary maps as input, and produces the coefficients of the homography matrix as output. The boundary distance is then computed by comparing the homography matrix to an identity matrix. Although the recently proposed boundary loss shows promising results for boundary refinement in segmentation tasks, the experiments in Chapter 3 had already been completed by the time this new boundary loss was published. Therefore, it is left for future studies on boundary refinement in material segmentation tasks.

2.4 Self-training

To utilise the boundary loss function, one fundamental requirement is that the labelled segments must fully cover the material area. As material segmentation datasets can be sparsely labelled, semi-supervised learning is a potential approach to fill in missing labels and provide dense segments that can be used with the boundary loss function. This approach utilises both labelled and unlabelled pixels during training. Among all semi-supervised learning approaches (Zhu, 2005), self-training is the most simple yet efficient one to fill in unlabelled pixels with generated pseudo labels. The illustration of this approach is shown in Figure 2.2. The idea is that the networks can train themselves with the pseudo labels generated by a trained network with existing samples. From the perspective of annotation, it can be considered a way of auto-labelling. If the initial network can identify a majority of the samples correctly, after repeating the process

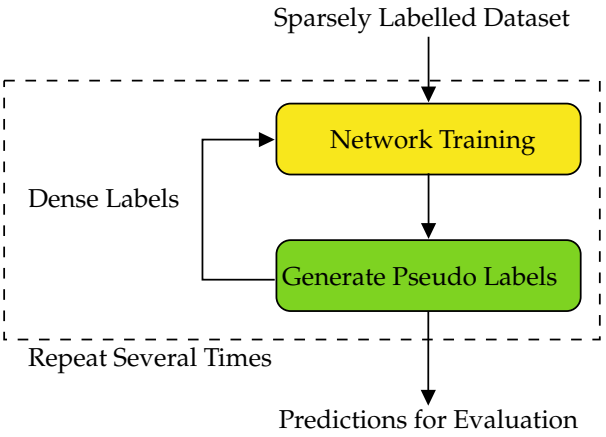


FIGURE 2.2: The pipeline of the self-training approach. It generates pseudo labels from a sparsely labelled dataset, and improves the pseudo label by repetitively training the network.

several times, the network performance should be improved. Recent experiments show that this approach can achieve SOTA segmentation performance with limited labelled samples (Le et al., 2015; Cheng et al., 2020; Zoph et al., 2020). In fact, even the well-known giant Segment Anything Model (SAM) (Kirillov et al., 2023) utilises such a self-training approach to enlarge its training set and improve network performance.

Although the self-training method may introduce more misclassified labels as noise to the dataset compared with more robust methods based on a discriminator to control pseudo label quality (Souly et al., 2017), the noise can also prevent the network from overfitting (Goodfellow et al., 2016, p. 241) since the LMD is a small dataset. Therefore, this thesis chooses the self-training method to generate pseudo labels and provide the boundary information for the CAM-SegNet. The experiments in Chapter 3 show that the self-training approach is not the factor that improves performance. Instead, the combined boundary and material features are the reason why the CAM-SegNet can perform well.

2.5 Transformers in Vision Tasks

The transformer is a type of neural network architecture that was first introduced in (Vaswani et al., 2017). It is composed of self-attention (Zhao et al., 2020b) and MLP (Tolstikhin et al., 2021) modules to learn the dependency between every pair of word tokens, and has become a very popular and powerful architecture in natural language processing (NLP), achieving SOTA results in a wide range of tasks, including machine translation, language modelling, and sentiment analysis (Devlin et al., 2018; Yu et al., 2021b).

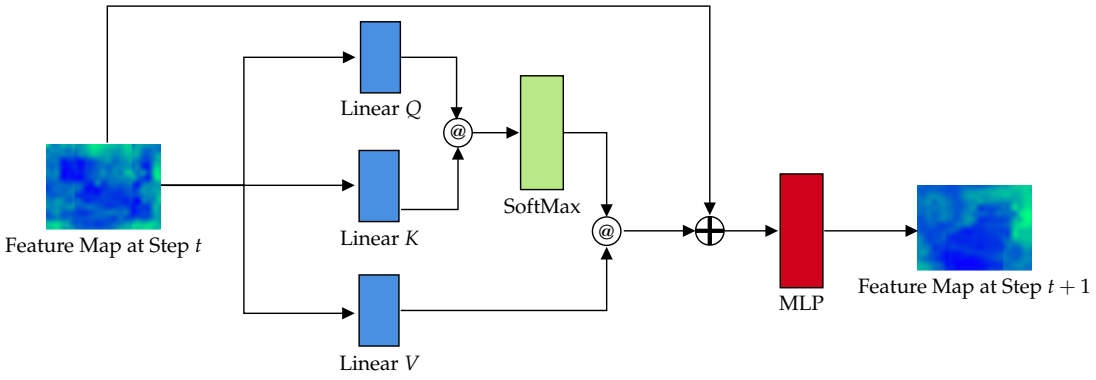


FIGURE 2.3: The architecture of a typical self-attention module.

In the computer vision domain, Dosovitskiy et al. first proposed to crop the images into 16×16 patches to train the Natural language processing (NLP) transformer architecture with images and achieved impressive performance in many classification datasets (Krizhevsky et al., 2012, 2009). The success of their work has demonstrated the potential of transformers in achieving promising performance in vision tasks. As a result, researchers are increasingly exploring the use of transformers in various computer vision applications such as classification (Dosovitskiy et al., 2020; Chen et al., 2021; Qing et al., 2021) and segmentation (Liu et al., 2021b; Strudel et al., 2021; Zheng et al., 2021), with the hope of achieving even better results than those obtained by traditional CNNs. Figure 2.3 illustrates the architecture of a typical transformer module. It uses three parallel linear (also known as fully-connected) layers to produce three matrices, named Query ($Q \in \mathbb{R}^{HW \times C}$), Key ($K \in \mathbb{R}^{HW \times C}$), and Value ($V \in \mathbb{R}^{HW \times C}$) respectively. The K^T and Q matrices are multiplied together, and then normalised with the SoftMax operation to generate the attention weights with dimensionality $\mathbb{R}^{HW \times HW}$ that each row represents the dependency for each pixel against all other pixels. The symbol @ represents matrix multiplication in this thesis, and H and W are the height and width of the feature map.

This section categorises vision transformers into global and local types based on the extent to which the self-attention module is applied. The global transformers represented by Vision Transformer (ViT) (Dosovitskiy et al., 2020) and DeiT (Touvron et al., 2021) employ global self-attention to capture the correlation between each pair of embedded patch features. This design ensures that such transformers have a global view from the first layer. However, the quadratic complexity in image size makes global transformers expensive to use. Moreover, a recent study (Raghu et al., 2021) showed that global transformers can still have a local view at shallow layers. Their work states that learning from local regions at the beginning is important for good performance. In contrast, local transformers such as Swin (Liu et al., 2021b,a) apply the self-attention module to windowed regions. This local design reduces the complexity to be linear in image size and gradually increases the patch size through patch merging. As a result, features from multiple stages of Swin are extracted from patches with different resolutions. By

aggregating these features through trainable weights, DBAT achieves the goal of learning from cross-resolution patch features.

2.6 Network Interpretability

The study of network interpretability aims to explain how a network makes predictions, and what features the network learns during training. For CNNs, the visualisation of the convolutional kernel weights shows the pattern of features that the network has extracted (Krizhevsky et al., 2012; Wang et al., 2020). For the attention-based network module, a simple yet effective way is to plot the per-pixel attention masks on the input image and the weights indicate the contribution to the final decision of each pixel (Fukui et al., 2019; Liu et al., 2018b). For transformers, however, interpreting the self-attention module remains challenging due to the high dimensionality of the correlation mask and the recursively connected attention modules. Carion et al. (2020) proposed to reduce the dimensionality by visualising an attention mask for individual pixels of the feature map one at a time. Chefer et al. (2021) reassigned a trainable relevancy map to the input image and propagate it through all the self-attention layers. However, these methods are designed for classification tasks and they can only interpret the transformer behaviour for a specific image. This thesis focuses on the segmentation task and prefers a summary explanation of the whole dataset. Therefore, this chapter chooses to introduce the CKA heatmap (Nguyen et al., 2020; Raghu et al., 2021) and the network dissection method (Bau et al., 2020, 2017, 2019). A detailed explanation is included in the following sections.

2.6.1 Centered Kernel Alignment

For transformers, the interpretability of the self-attention module remains challenging due to its high dimensionality and recursive network connections. Carion et al. (2020) proposed to reduce the dimensionality by visualising one self-attention layer for a single pixel at a time. Chefer et al. (2021) reassigned a relevancy map to the input. The map propagates through all the self-attention layers to capture the network structure. However, these methods can only illustrate the transformer behaviour for a specific input image. To obtain a summarised explanation across the whole dataset, Section 5.2 plots the CKA matrix (Nguyen et al., 2020; Raghu et al., 2021; Kornblith et al., 2019) which measures the similarity between two layers from network layer features evaluated with the same group of samples.

In this section, $X \in \mathbb{R}^{m \times d_1}$, $Y \in \mathbb{R}^{m \times d_2}$ are the feature maps extracted from two network layers. Here m is the number of samples and d_1, d_2 are the dimensions of the flattened feature map. To understand the equations of CKA, it is essential to introduce the vector

similarity and the Gram matrix. For vectors a, b, the cosine similarity is defined as the dot product between a, b divided by the product of their norms, $\frac{<a,b>}{|a||b|}$. Here $< \cdot >$ is the dot product. The cosine similarity is within the range [-1,1], where 1 indicates that vectors a and b are exactly the same in terms of direction and length, value -1 indicates that they have the same length but in opposite direction, and value 0 means that they are orthogonal and uncorrelated to each other. The vector similarity measurement is then extended to the matrix, known as the Gram matrix with shape $m \times m$ in Equation 2.6, where the element at position i, j represents the dot product between sample i and sample j. In this way, the measurement of the similarity between two arbitrary layers can be transformed into measuring the similarity of two Gram matrices by Equation 2.7, regardless of the feature map resolution. Here vec flattens the matrix into a vector, and $\| \cdot \|_F$ is the Frobenius norm.

As for the CKA matrix, it is a general form of Equation 2.7 where the positive definite kernel function is applied to calculate the Gram matrix elements as $k =< \phi(x_i), \phi(x_j) >$), $l =< \Phi(y_i), \Phi(y_j) >$, and the Gram matrix is centered as K', L' by projecting K, L onto the space orthogonal to the vector $\mathbf{1}$ through HKH, HLH, where H is the centering matrix $H = I_m - \frac{1}{m}\mathbf{1}\mathbf{1}^T$, where I_m is an identity matrix of size $m \times m$ and $\mathbf{1}$ is a vector of length m and all elements are 1. The kernel function is used to deal with more complex relationships between the features. The Gram matrix ensures that the CKA matrix is invariant to orthogonal transformations.

In this thesis, the CKA matrix measures the layer similarity by normalising the Hilbert-Schmidt Independence Criterion (HSIC) (Song et al., 2012), as shown in Equation 2.8 (Raghu et al., 2021). By normalising HSIC with Equation 2.9, the CKA becomes invariant to isotropic scaling. Since HSIC$= 0$ only when X and Y are independent and CKA$= 1$ when X and Y are the same, they together give a meaningful comparison of two networks with different architectures (Kornblith et al., 2019).

$$K = XX^T, L = YY^T \tag{2.6}$$

$$feature\ similarity = \frac{< \text{vec}(K), \text{vec}(L) >}{\|K\|_F \|L\|_F} \tag{2.7}$$

$$\text{HSIC}(K, L) = \frac{< \text{vec}(K'), \text{vec}(L') >}{(m-1)^2} \tag{2.8}$$

$$\text{CKA}(K, L) = \frac{\text{HSIC}(K, L)}{\sqrt{\text{HSIC}(K, K)\text{HSIC}(L, L)}} \tag{2.9}$$

As a model-independent method, the CKA matrix enables the quantitative comparison between two networks regardless of their architectures. This thesis illustrates the behaviour of the DBAT by computing the CKA matrix of itself and its backbone transformer. It shows that DBAT learns new features from the aggregated cross-resolution

patch features to improve performance. Following (Kornblith et al., 2019; Nguyen et al., 2020), the unbiased estimator of HSIC (Song et al., 2012) is used in this thesis.

2.6.2 Network Dissection

Visualisation tools and CKA help to understand how the model combines specific features to predict material labels and highlight the similarity or independence of features acquired by different network layers. However, they do not offer insight into the nature of these features. To address this issue, this thesis utilises the technique from the 'network dissection' literature (Bau et al., 2017; Zhou et al., 2018; Bau et al., 2019, 2020), which correlates neuron outputs to an independent set of human-interpretable labels, such as objects, textures, or scenes. It treats the features extracted by one neural unit (corresponding to one channel in the feature map) as the segmentation solution and measures the correlation with the ground truth segments in each semantic category. If the correlation exceeds a pre-defined threshold, then the neuron is considered to learn features related to the corresponding semantic label.

Calculating the correlation score requires a densely labelled dataset containing labels for a set of pre-defined concepts. In this thesis the Broden dataset proposed by Bau et al. (2017) is used to interpret the networks. First, the trained parameters of a network are frozen. Then, the output of each neuron in the last network layer is thresholded into a binary mask to be compared with the corresponding concept ground truth segments in terms of mIoU (Bau et al., 2017). The threshold is the value a_k ensuring that 99.5% of the activation values are greater than it. A neuron is assigned the interpretive label for which the mIoU score is the highest and above 0.04. By measuring the number of neurons aligned with each concept, the network dissection method indicates the features the network focuses on during training.

The network dissection method is applied in Chapter 5 to compare the proposed DBAT with selected networks. The results show that the DBAT is particularly good at detecting local material features, such as texture, which may be the reason why DBAT achieves the narrowest uncertainty bound when the network is trained independently for five times.

2.6.3 Interpretable Networks

It is worth noting that the network dissection method can only interpret disentangled neurons (Bau et al., 2020; Shen et al., 2021). This means that only a fraction of the channels of a network layer can be aligned with meaningful semantic concepts. The rest of the neurons also detect useful features, but the features that they learn cannot be explained with semantic labels. One of the reasons is that these neurons are detecting

mixed features (*e.g.*detecting both texture and object combinations). In order to reduce the number of entangled neurons, the concept 'Interpretable Networks' is proposed. The idea is to disentangle the patterns that each neuron learns so that the visualisation of the feature map becomes interpretable. Zhang et al. (2018a) proposed to separate the patterns that a network learns by building an explanatory graph. The explanatory graph can be applied to trained networks and summarises the extracted features into a few patterns. Zhang et al. (2018b) further introduced a filter loss term to regularise the features so that each neuron contributes to one category with one consistent visual pattern. However, their networks can only learn features from ball-like areas since the filter loss is based on a regional template. Shen et al. (2021) extended the interpretable networks to learn disentangled patterns without shape or region limitations. Their compositional network splits neurons into groups and makes the neurons learn similar/different features within/across the groups. The trained networks can produce meaningful feature maps with a slight sacrifice in accuracy (Shen et al., 2021). However, training an interpretable network is beyond the scope of this research and shall be investigated in the future.

2.7 Material Property Measurements

Unlike the properties of objects, which are often associated with appearance semantics like shape and colour, the properties of materials are often defined by how they interact with light or sound at their surface and require specialised equipment for measurement, in addition to using adjectives to describe the visual or tactile properties of a material (Schwartz and Nishino, 2019). Portable measurement devices such as Time-of-Flight (ToF) cameras (Su et al., 2016) and hyperspectral cameras (Behmann et al., 2018) have the capability to assess the reflective or scattering properties of materials. ToF cameras operate as indirect sensors by determining the elapsed time for a light pulse to travel from the camera to the material and back (Su et al., 2016), while hyperspectral cameras directly capture the spectral signature, which quantifies how much light can be reflected by the material at sampled wavelengths (Grewal et al., 2022). For material segmentation, hyperspectral cameras are preferred due to their ability to capture a complete scene and provide a comprehensive measurement of material properties through their spectral profiles.

Apart from mobile devices, laboratory devices can measure spectral information under a constrained environment. For example, the spectrophotometer (Albert et al., 2012) quantitatively measures the absorptance and reflectance distribution against visible and infrared radiation wavelengths based on the amount of light absorbed by the material (Lv et al., 2022; van Nijnatten, 2014). The streak camera (Bagayev et al., 2020; Horn, 2009) measures time-dependent temporal point spread functions (TPSF) (Kirkby and Delpy, 1996), which describe how light is reflected, refracted, scattered, or absorbed

by the material. Additionally, femtosecond lasers (Lureau et al., 2020) can measure the thermal conductivity and mechanical properties of the material by emitting pulses of light and performing time-resolved measurements of temperature and laser-induced deformations (Guo et al., 2019). A spectrophotometer and a spectral camera are two devices that measure spectral information, but differ in their measurement capabilities. A spectrophotometer measures the amount of light absorbed or transmitted by a sample at a single point or small area, making it a precise tool for material characterisation and analysis in laboratory settings. In contrast, a spectral camera captures spectral information for an entire scene, making it useful for remote sensing and imaging applications. The samples in the spectraldb (Jakubiec, 2022) are measured by a spectrophotometer, and samples in the ARAD_1K (Arad et al., 2022) are measured by a spectral camera. The spectral profile can facilitate a correlation between measurements obtained from hyperspectral cameras and spectrophotometers provided that the measurements from the hyperspectral camera are lighting invariant. MatSpecNet proposed in Chapter 6 leverages both datasets by matching the measurements with a similarity score that describes the shape difference of two spectral profiles.

In addition to sensor-based measurements, human perception also plays a crucial role in evaluating material properties. For instance, the photopic reflectance V(λ) and melanopic reflectance M(λ) are derived from the measured spectral profile based on the human visual system. The photopic reflectance captures the average human response to the brightness of light in the visible spectrum (Smith and Pokorny, 1996), while the melanopic reflectance provides information about the effect of reflected light on the activity of melanopsin photoreceptors in the human eye (Lucas et al., 2014). Moreover, the roughness or irregularity of material surfaces, which are difficult to measure with devices, can be estimated through appearance-driven assessments based on human observation (Jakubiec, 2022; Jones and Reinhart, 2017).

In Chapter 6, MatSpectNet employs the ARAD_1K dataset (Arad et al., 2022), captured by a hyperspectral camera, as the training data for the spectral recovery network, $S(x)$. To refine the precision of the recovered hyperspectral images and incorporate human observations, the spectral and observation measurements in the spectraldb (Jakubiec, 2022) are utilised, which are acquired from a spectrophotometer, as a correction reference.

2.8 Material Segmentation in Remote Sensing

In the expansive realm of computer vision, the concept of material segmentation finds application in diverse contexts, including remote sensing through satellite and airborne imagery. This technique, although not novel and has garnered over decades of usage

(Grewal et al., 2023), continues to shape our understanding of Earth surface by harnessing advanced technologies such as hyperspectral images and neural networks.

While the material definition in remote sensing, encompassing domains like agriculture and forestry (Demir et al., 2018), as well as specific instances such as Corn and Grass-trees (Baumgardner et al., 2015), differs from the realm of indoor material segmentation, their segmentation methodologies offer valuable insights that illuminate my own research.

In the previous decade, colleagues have grappled with challenges such as imbalanced and limited training data, as well as the intricacies of the curse of dimensionality. Preceding the advent of deep learning, hyperspectral image segmentation in remote sensing relied on factors such as colour, pixel intensity, texture, and an array of diverse features. For example, threshold-based segmentation is a simple yet effective image-processing technique used to separate objects or regions of interest. This method involves setting a threshold value and classifying pixels or regions based on their intensity levels relative to the threshold. Ghamisi et al. (2012, 2013) proposed a multi-level thresholding method that leverages the natural computing method named particle swarm optimisation (Mirjalili and Mirjalili, 2019). Wu et al. (2018) introduced an adaptive threshold segmentation technique to address the issue of manually selecting empirical threshold values. The threshold was determined by computing the mean distance between training samples and the spectral centre of each class.

In addition to the threshold-based segmentation method, clustering (Verma et al., 2016; Chen et al., 2011; Pisani et al., 2014), edge detection-based segmentation (Youn and Lee, 2013; Xia et al., 2016) and other computer vision techniques (Angulo et al., 2009; Li et al., 2018, 2019) have been investigated for hyperspectral image segmentation as well. Chen et al. (2011) proposed a method involving the utilisation of multiple kernel fuzzy c-means clustering for hyperspectral image segmentation. Verma et al. (2016) employed an enhanced approach using intuitionistic fuzzy c-means clustering that incorporated local spatial information for each pixel, retained visual features, and exhibited resistance to noise. Angulo et al. (2009) utilized a multi-scale stochastic watershed approach to achieve unsupervised segmentation of the hyperspectral images. To incorporate edge information, Youn and Lee (2013) employed the Bhattacharya distance to assess the similarity between adjacent blocks, and Xia et al. (2016) applied edge-preserving filtering to spectrally independent components selected by independent component analysis.

With the rise in popularity of deep learning methods, the hyperspectral segmentation task can be solved using proper network architectures and training strategies. Akiva et al. (2022) adopted a self-supervised pre-training strategy to learn robust features for remote sensing material segmentation. Zhou et al. (2019b) proposed a Long Short Term Memory (LSTM) network to learn both spectral and spatial features. Li et al.

(2023) chose to embed manifold subspace learning and learn from intrapatch samples to model unified spectral–spatial feature representations. As for the proposed MatSpectNet, the idea of combining spectral-spatial features is also used by adopting the MLP to learn material features from hyperspectral images.

2.9 Hyperspectral Image Recovery Methods

Early attempts to recover hyperspectral images rely on sparse coding methods, such as manifold representation, which embeds high-dimensional spectral information into low-dimensional representations (Li et al., 2020; Jia et al., 2017). Recent network-based methods investigate network modules that learn both spatial and spectral features (Cai et al., 2022c; Hu et al., 2022). While these methods have achieved accurate spectral recoveries, the application of hyperspectral images is still limited by the lack of annotated hyperspectral image datasets with semantic labels. Despite advancements in solving the spectral recovery challenge for geoscience applications such as aerial image dehazing, the challenge remains a topic of ongoing research for daily images (Mehta et al., 2021; Cai et al., 2022b; Arad et al., 2022). The method in Chapter 6 investigates how to apply hyperspectral recovery methods to existing material segmentation datasets. The proposed method can also be applied to other tasks without much modification.

2.10 Physically Based Rendering

The purpose of PBR is to synthesise realistic images from 3D scene descriptions with physics principles, which models the interaction of light and materials (Pharr et al., 2016). The well-known ray-tracing algorithm (Pharr et al., 2016; Meister et al., 2021; Quatresooz et al., 2021) is the foundation of modern render engines such as Mitsuba (Nimier-David et al., 2019) and Nvidia OptiX (Ludvigsen and Elster, 2010; Rott, 2022). The algorithm traces the lights arriving at the camera from the light source, with the consideration of how light interacts with the materials of scene objects. The interaction can be described as functions that calculates how much energy of the light is reflected, *e.g.*the Bidirectional Reflectance Distribution Function (BRDF) (Sun and Zhao, 2021). Since these functions are closely related to material optics properties such as absorption and reflection, the 3D scene descriptions should include detailed material definitions. One possible solution to create a material segmentation dataset is to assign material descriptions from ambientCG (Demes), which provides free PBR materials, to 3D objects from 3D-FRONT (Fu et al., 2021), and renders the images in Mitsuba (Nimier-David et al., 2019). To ensure material diversity, the texture of each material description can be randomly selected to cover various appearances. However, existing open-access material descriptions are limited to supporting the render engine in synthesising RGB

images. Collecting material descriptions that can generate hyperspectral images is a non-trivial task. Thus, this thesis considers PBR as a potential future research direction in the realm of dense material segmentation.

Chapter 3

CAM-SegNet: A Context-Aware Dense Material Segmentation Network

This chapter presents the contributions of my first-year PhD study, including a hybrid network called CAM-SegNet and its training strategies. The network is designed to learn the material and contextual features during training jointly. It can accomplish this task even when trained with sparsely labelled material segmentation datasets such as LMD (Schwartz and Nishino, 2016, 2020), without requiring additional object or scene annotations. This chapter will first outline the motivation and research question, followed by introducing the proposed methodology. Next, the experiment design and results will be presented. Finally, the contribution will be summarised, including the benefits and drawbacks of the approach.

3.1 Research Question and Motivation

The research question of how to train a network to learn both material and contextual features simultaneously without relying on additional contextual annotations is addressed in this chapter. According to Schwartz and Nishino (2020, 2016), combining material and contextual features can enhance network performance in material segmentation tasks (Schwartz, 2018; Schwartz and Nishino, 2020, 2016; Bell et al., 2015b). They indicate that material features enable the network to identify material categories despite their varying appearances, while contextual features constrain the possible material categories that appear in a given scene. To investigate this assumption, they present a multi-branch network architecture (Zhang et al., 2020b, 2019b), which includes one branch for extracting material features from image patches, and multiple

pre-trained branches targeting object segmentation and scene recognition tasks to extract contextual features. The material and contextual features are then concatenated to predict material labels. Although their work offers a viable approach to achieve dense material segmentation with a neural network, the network design still requires refinement since the pre-trained branches that provide contextual features are not fine-tuned along with the material branch, given that dedicated material datasets lack contextual labels.

3.2 Overview

This chapter presents the CAM-SegNet, a hybrid network architecture that enhances the performance of dense material segmentation. Unlike previous approaches that adopt disentangled material and contextual features, the CAM-SegNet shows that a carefully designed mechanism to combine these features during training can lead to better segmentation performance. The network is composed of global, local, and composite branches. The global branch extracts contextual features from the whole image, while the local branch learns material features from image patches. The composite branch then generates the material predictions by merging the features. The effectiveness of the CAM-SegNet is demonstrated by adjusting the global branch to extract boundary-related contextual features, using a loss function that measures the alignment between the predicted and ground truth material boundaries.

To address the weakness of sparsely labelled datasets, a self-training approach is employed to augment the existing segments with predicted pseudo labels. The performance of the proposed CAM-SegNet architecture is evaluated on both the sparse LMD test set and the DLMD test set, which includes eight indoor scene images with dense annotations. The results show that CAM-SegNet outperforms recently proposed network architectures and single-branch approaches in the control group by 3-20% in Pixel Acc and 6-28% in mIoU. Furthermore, the iterative self-training process does not compromise the accuracy of the CAM-SegNet.

3.3 CAM-SegNet Architecture

This section presents the overall network structure, as illustrated in Figure 3.1. The global branch takes the original image as input while the cropped patches are fed into the local branch. The encoders extract features from both branches independently and downsample the feature maps. The decoders recover the feature map resolution jointly (with the feature sharing connection) and generate the outputs for each branch. The composite branch crops and concatenates the global branch output O_G to the local

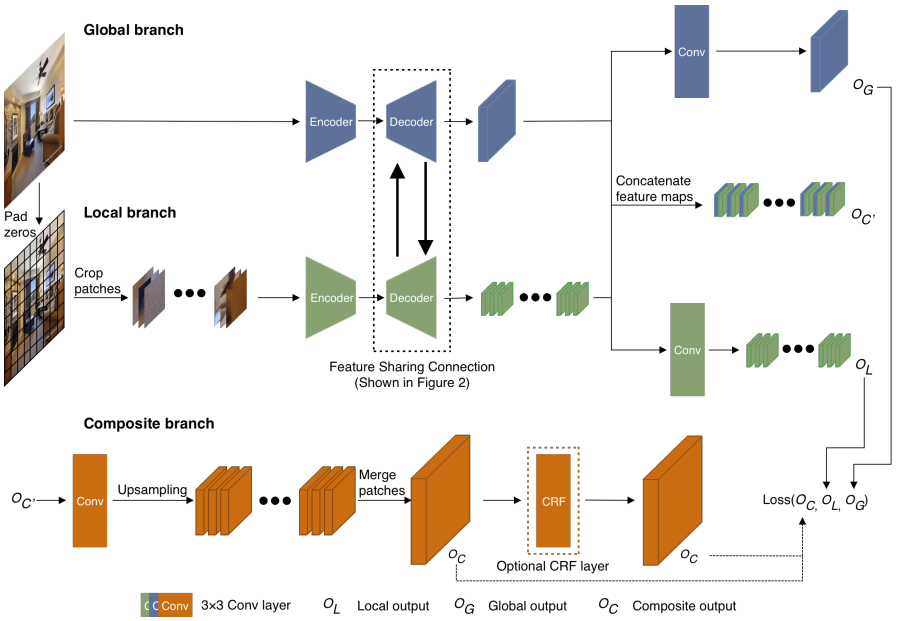


FIGURE 3.1: **CAM-SegNet** architecture. The feature maps in the decoders are shared between the global and local branches. After the encoder-decoder component, the feature maps at the same spatial location are concatenated together and passed into the composite branch, which upsamples the feature maps to the same size as the original input image. The composite output can be refined by an optional CRF layer.

branch output O_L. Then the network merges the upsampled feature maps, and generates the composite output O_C. The last convolutional layer is applied to patch feature maps $O_{C'}$, to ensure that the overall network still focuses on material information extracted from image patches. Finally, the optional CRF layer can be used to refine the composite output O_C. While training, the contextual features extracted from the global branch are controlled by the loss function applied to the global branch output O_G. When inferring unseen images, only the composite output O_C is kept to generate the final segmentation. To ensure that the matched global and local feature maps are learning from the same image region, the patch cropping method used to crop the input images is adopted, as described in Algorithm 1. The feature merging process is precisely the reversed cropping operation, and the overlapping pixels between patches are averaged.

3.3.1 Feature Sharing Connection

The decoder in Figure 3.1 gradually upsamples the feature maps with three convolutional blocks. To train the two branches collaboratively, at the input of each block, the feature maps are shared between the global and local branches through the feature sharing connection showed in detail in Figure 3.2. The feature maps are defined as $X_G \in \mathbb{R}^{c \times h_G \times w_G}$ for the global branch, and $X_L \in \mathbb{R}^{b \times c \times h_L \times w_L}$ for the local branch.

Algorithm 1
This algorithm is designed to calculate the parameters when cropping the input images or feature maps. The same parameters are used to merge the patches to ensure the feature value at the corresponding position describes the same image region in the global and the local branch.

procedure GETPATCHINFO(*PatchSize*, *S*) ▷ *S* is the height or width of the original image
  Initialize
    num_patch ← 0 ▷ Number of patches cropped alone one dimension
    stride ← 0 ▷ Number of pixels to next patch
    pad ← 0 ▷ Number of zeros to pad at a particular dimension
  if *S* **mod** *patch_size* **equal** 0 **then** ▷ When the patches accurately cover the image
    num_patch ← *S* **divide** *patch_size*
    stride ← *patch_size*
  else ▷ Allow padding and overlapping for one more patch
    num_patch ← (*S* **divide** *patch_size*) **plus** 1
    stride ← (*S* **divide** *num_patch*) **plus** 1
    pad ← (*stride* **multiply** (*num_patch* **minus** 1)) **plus** *patch_size* **minus** *S*
  return *num_patch*, *stride*, *pad*

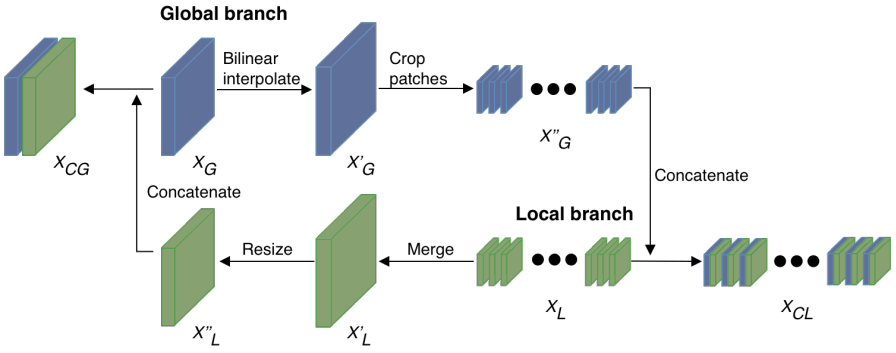


FIGURE 3.2: The feature sharing connection between the decoders. X_{CG} is the concatenated global branch feature maps, while X_{CL} is the concatenated local branch feature maps.

Here c represents the channel number, h, w are the height and width, and b is the number of patches. First, the global branch feature maps X_G are cropped into patches, $X'_G \in \mathbb{R}^{b \times c \times h_L \times w_L}$, and these patches are concatenated with the local branch feature maps. Then the network merges the patch feature maps X_L from the local branch to produce $X'_L \in \mathbb{R}^{c \times h_G \times w_G}$. Finally, the merged feature maps are concatenated with the global branch feature maps. The number of channels in the concatenated feature maps, X_{CG} and X_{CL}, are doubled to $2c$. To ensure that the global and local feature maps can match each other spatially, the same patch cropping method is used as the one used to crop the input images.

3.3.2 Context-Aware Dense Material Segmentation

The three outputs (O_G, O_L, O_C) generated from the CAM-SegNet make it convenient to control the features extracted from each branch, by optimising the branches to achieve different tasks with different loss functions. To optimise the CAM-SegNet, the total loss function L_{total} can be defined as

$$L_{total} = L_{global}(O_G, Y_{1/4}) + L_{local}(O_L, Y_{1/4}) + L_{composite}(O_C, Y) \tag{3.1}$$

where Y is the ground truth segment, and $Y_{1/4}$ is the downsampled ground truth segment. The downsampled ground truth is used to reduce the memory capacity needed during training. This chapter aims to combine contextual and material features to generate dense material segmentation. According to Schwartz and Nishino (2016, 2020); Schwartz (2018), material patches without contextual cues can force the network to extract material features. Since the local branch is responsible to learn from image patches, it is optimised to provide material features with the focal loss (Lin et al., 2017b), *i.e.*, $L_{local} = L_{focal} = \frac{1}{N}\sum_i -(1-p_i)^3 log(p_i)$. Here N is the number of pixels in O_L, and p_i is the estimated probability of pixel i in O_L for the true category. Similarly, the global branch is optimised to provide contextual features since the original images contain contextual information. However, contextual labels (*e.g.*objects or places) are needed to extract corresponding contextual features. Although these features can reduce the material segmentation uncertainty (Schwartz and Nishino, 2016), the cost of extra labels is not desired.

Instead of exploring contextual features that need extra labels, the CAM-SegNet investigates the contextual information that is missing in the image patches — the boundary between different materials. For pixels along the boundary of material c, let R^c, P^c be the recall and precision score. To provide boundary related features, the boundary loss (Bokhovkin and Burnaev, 2019), $L_{global} = L_{boundary} = \sum_c 1 - \frac{2R^c P^c}{R^c + P^c}$, is applied to the global branch output O_G, which aligns the predicted material boundary with the ground-truth segments. Ideally, the composite branch should be able to generate predictions accurately with good boundary quality, if the composite branch can learn from the outputs from both branches properly. Therefore, the composite branch loss function $L_{composite}$ is set as $L_{boundary}(O_C, Y) + L_{focal}(O_C, Y)$ to ensure that it is optimised to achieve these two goals at the same time.

3.3.3 Self-Training Approach

Since not all training segments in LMD cover the whole material region, the detected ground truth boundaries may provide misleading information to the boundary loss

(Bokhovkin and Burnaev, 2019). Therefore, the first task is to complete the labels. A network is trained with the focal loss (Lin et al., 2017b) and sparsely labelled LMD as the initial teacher model to generate pseudo labels. It is assumed that the LMD augmented with pseudo labels can provide necessary boundary information for the CAM-SegNet. The teacher-student-teacher self-training approach (Zoph et al., 2020) contains four stages:

1. The initial teacher model is trained by setting all the loss terms in Equation 3.1 to L_{focal}.
2. The trained teacher model generates the feature maps for the training set, and replaces the known pixels with ground truth labels.
3. The feature maps are refined by the CRF layer, to produce the final pseudo labels with better material boundary.
4. A CAM-SegNet is trained as a student model with the augmented LMD.

To improve the pseudo label quality and achieve the best performance, typical self-training cases such as (Zoph et al., 2020; Le et al., 2015; Cheng et al., 2020) repeat this training approach many times, to produce a series of student models. In detail, the student model at round t is considered as the new teacher model, to produce a new augmented dataset with the second and third stages. Then this dataset is used to produce a new student model, S_{t+1} with the fourth stage. It is worth noting that, the self-training strategy may not work well if the initial teacher model cannot predict most of the labels correctly. According to Bank et al. (2018), an initial accuracy of 70% is not enough. Since the reported material segmentation accuracy is about 70% in (Bell et al., 2015b; Schwartz and Nishino, 2016, 2020), it is not expected to achieve an increased accuracy. Instead, the objective is to show that the additional boundary information can help the network to generate segments with good boundary quality, and the self-training approach is one way to provide such information.

3.4 Experiments

This section introduces the experiment configurations to train the CAM-SegNet, including dataset pre-processing methods and evaluation metrics. The recently proposed models in the literature are chosen as baseline models to show the superiority of the CAM-SegNet.

3.4.1 Dataset

The experiments evaluate the proposed method on the LMD (Schwartz and Nishino, 2016, 2020), and follow their suggestion to crop the images into 48 $\times$ 48 patches. The samples are randomly split into training (70%), validation (15%) and test (15%) sets. Since the contributions mainly focus on indoor material segmentation, this chapter presents qualitative evaluations of the segmentation results only for images taken in indoor scenes such as kitchens and living rooms.

3.4.2 Evaluation metrics

The network performance is evaluated with the Pixel Acc and the mIoU score. It is worth pointing out that the sparsely labelled segments in LMD may not reflect the true segmentation quality, especially for pixels near the material boundaries. Therefore, in addition to the LMD test set, eight indoor images are exhaustively labelled in the LMD test set to evaluate the performance of the proposed network. These eight images are referred as DLMD in the experiments.

In addition to the material segmentation performance, the metrics related to network efficiency, such as the number of parameters (#params), the number of floating point operations (#flops) and the frames per second (FPS). The #params refers to the adjustable weights in a neural network, impacting its capacity to learn complex patterns. The #flops represents the computational workload during a forward pass, influencing processing efficiency. Regarding FPS, while it typically demonstrates an inversely proportional relationship with the #flops within the same network architecture, this correlation might not persist across diverse architectures. The reason is that FPS can be influenced by factors beyond network components, including operations like image pre-processing and post-processing, clipping and merging, which can impact overall FPS.

3.4.3 Baseline Models

The main contribution of this chapter is to combine both global contextual features and local material features to achieve dense material segmentation. To show the advantage of the proposed model among SOTA networks for image segmentation task, DeepLabV3+ (Chen et al., 2018), BiSeNetV2 (Yu et al., 2020), and PSPNet (Zhao et al., 2017b) are selected as the baselines. In the experiments, the pre-trained models implemented by (Yakubovskiy, 2020) are fine-tuned. Since these networks have not been evaluated on the LMD previously, the training procedures from their original papers are adopted and the same backbone (ResNet-50) are used as the CAM-SegNet. The results are refined by the same CRF layer for a fair comparison.

3.4.4 Implementation details

The ResNet-50 (He et al., 2016) pre-trained on ImageNet (Deng et al., 2009) is used as the encoder and the FPN (Lin et al., 2017a) is used as the decoder. The skip connections are added between the encoder and decoder as in (Chen et al., 2019). The patch size 48 is not divisible by the default encoder downsampling factor 32, which may cause a spatial mismatch between the local and global feature maps. Therefore, the downsampling factor is changed to 16, by setting the stride of the final block convolutional layer to 1. Since Schwartz and Nishino (2016, 2020) did not release the segmentation task training configuration, this section follows the work in (Bell et al., 2015b) to normalise the images by subtracting the mean (124, 117, 104) for the RGB channels respectively. To refine the segmentation outputs, the trainable Conv-CRF (Teichmann and Cipolla, 2019) is adopted. First, the Adam optimiser with learning rate 0.00002 is used to train the network without a CRF layer. Then the network parameters are frozen to train the CRF layer with learning rate 0.001. Finally the network is refined together with the CRF layer with learning rate 0.0000001. Each stage is trained for 40 epochs. Since the images have different sizes, the gradients are accumulated to achieve an equivalent batch size of 32. According to Chen et al. (2019), a mean squared error regularisation term between the global and local outputs can help the network to learn from both branches. This regularisation term is removed when the CRF layer is appended to the network, to encourage the branches to learn more diverse features. The self-training approach is repeated three times.

3.5 Result Analysis

This section presents the quantitative and qualitative evaluations, including the comparison between CAM-SegNet and baseline models, and the ablation study that tears apart the network to validate each component of the architecture.

3.5.1 Quantitative Evaluation

Table 3.1 compares the performance of the CAM-SegNet against the baseline models. CAM-SegNet achieves comparable performance compared with DeepLabV3+ on the LMD. When evaluated on the DLMD, the CAM-SegNet achieves 3.25% improvement in terms of Pixel Acc and 27.90% improvement in mIoU, compared with the second highest score achieved by DeepLabV3+. This indicates that the proposed CAM-SegNet successfully learns the difference between materials and precisely predicts points near the boundary. Moreover, the improved performance demonstrated on the DLMD suggests that evaluating segmentation models solely on sparsely labelled datasets may not provide a reliable estimation of their actual performance. In order to illustrate

the model performance for individual materials, seven common materials that exist in indoor scenes from DLMD are chosen to report the per-category Pixel Acc values. DeepLabV3+, BiSeNetV2 and PSPNet got low scores on materials of small objects, such as foliage (plants for decoration) and paper. Another observation from Table 3.1 is that these three networks can still achieve comparable performance when recognising materials that usually cover a large area of the image, such as plaster (material of the wall and ceiling) and wood (usually wooden furniture).

One reason for the low scores may be that the networks failed to learn from local material features such as texture. PSPNet relies on the pooling layers to learn from multi-scale features, DeepLabV3+ uses dilated convolutional layers. Although BiSeNetV2 adopts two branches to learn from local and global features, they all take the full-size images as input, and the intermediate layers do not communicate during training. The local features can fade out especially when the image resolution is low. As a consequence, these networks tend to depend on global features and may not recognise small material regions well.

In contrast, the CAM-SegNet adopts both full-size images and cropped patches, to learn from the global and local features, which are combined and co-trained. This enables the CAM-SegNet to recognise materials that are hard to identify (foliage and paper) for the baseline models.

Models	ceramic	fabric	foliage	glass	paper	plaster	wood	Pixel Acc		mIoU	#params (M)	#flops (G)	FPS
Pixel Coverage (%)	2.95	10.96	15.43	1.94	1.76	2.54	13.54	-		-			
Datasets				DLMD				LMD	DLMD	DLMD			
DeepLabV3+	**97.68**	27.56	0.00	48.91	0.00	**88.94**	73.69	71.37	67.09	32.04	59.57	51.44	21.55
BiSeNetV2	18.86	3.07	0.00	23.00	0.34	58.68	70.77	45.66	37.66	15.08	3.40	21.15	156
PSPNet	55.59	0.12	0.00	**66.73**	1.47	79.25	73.76	50.12	52.11	23.39	65.73	68.19	5.44
CAM-SegNet (ours)	92.65	**32.72**	**88.81**	21.99	**30.67**	87.77	**93.82**	**71.65**	**69.27**	**40.98**	52.31	56.24	13.25

TABLE 3.1: Quantitative evaluation results for the CAM-SegNet and baseline models. The values are reported as percentages. The highest value for each evaluation metric is in bold font. Seven common indoor materials are selected to report the performance of Pixel Acc on the DLMD. The number after the material category is the pixel coverage (in percentage) of each material in the dataset. The Pixel Acc is evaluated on both LMD and DLMD. Since LMD test set provides sparsely labelled images, it is not meaningful to report mIoU on LMD. Therefore, mIoU is reported on DLMD only.

3.5.2 Qualitative Evaluation

Figure 3.3 compares the segmentation quality of the CAM-SegNet with the DeepLabV3+. As indicated by the mIoU score, CAM-SegNet is better at recognising pixels around material boundaries. In the kitchen image, the boundary between the ceramic floor and the wooden cupboard is adequate. In the toilet image, the ceramic close-stool is successfully separated from the wall covered with plaster. A more detailed qualitative evaluation is included in the ablation study.

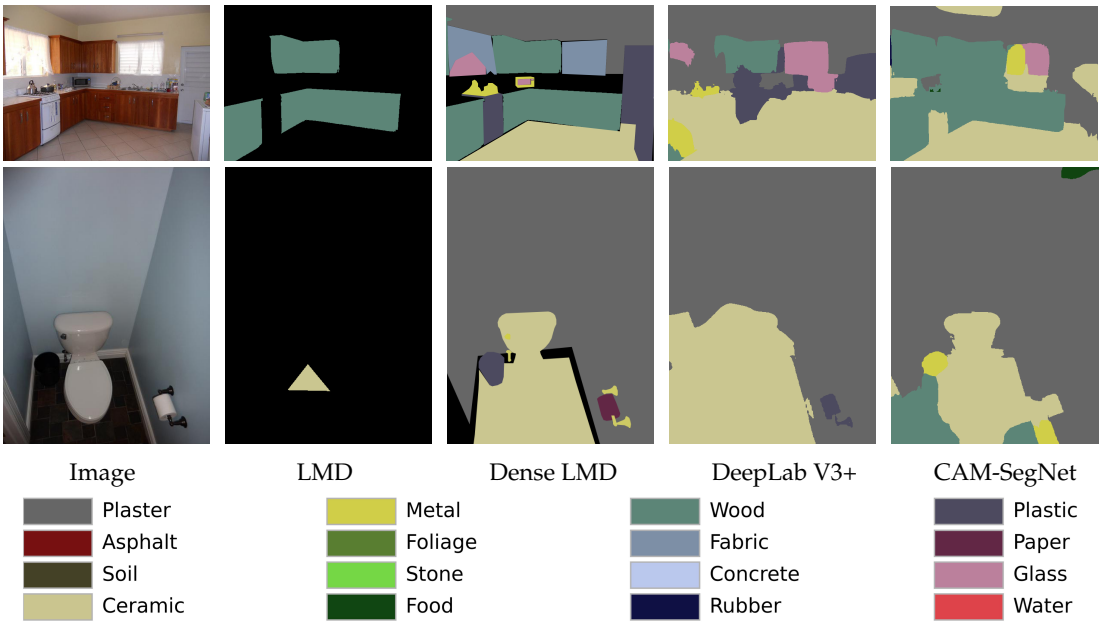


FIGURE 3.3: Dense material segmentation results for Kitchen image and Living Room image. The sparsely labelled images are taken from LMD, and densely labelled with all known material categories manually.

3.5.3 Ablation Study

Table 3.2 and 3.3 evaluate the effectiveness of each component of the CAM-SegNet. The components include the network architecture, the loss function, and the CRF layer. For fairness, all models are trained with the same training procedure as the CAM-SegNet. In detail, to show the advantages of the two-branch architecture, two single-branch models are trained with full-size images and image patches separately, and are referred to as the Global and Local models respectively.

Since the LMD is sparsely labelled, it is not straightforward to train the proposed CAM-SegNet without the self-training approach. In order to control for the influence of the self-training approach, the CAM-SegNet is retrained with the focal loss (Lin et al., 2017b). The loss is applied to all three outputs in Equation 3.1, and the trained model is named as the Self-Adaptive CAM-SegNet (SACAM-SegNet). To avoid confusion, the CAM-SegNet trained with the boundary loss is referred to as the Boundary CAM-SegNet or BCAM-SegNet. Table 3.2 shows that the SACAM-SegNet achieves an improvement of 12-20% on Pixel Acc and 6-19% on mIoU, compared with single-branch models without the self-training approach. Although PAC-CRF refined models tend to get higher Pixel Acc, Conv-CRF refined models can achieve higher mIoU.

Figure 3.4 shows that the SACAM-SegNet can produce correct labels for pixels that are hard to recognise for the Global or Local models. For example, the SACAM-SegNet can label the window in the kitchen as glass correctly. Moreover, the proposed model can

Metric	CRF Layer	Local	Global	SACAM-SegNet
Pixel Acc	PAC-CRF	61.95	60.58	**69.25**
	Conv-CRF	58.07	55.67	**66.83**
mIoU	PAC-CRF	27.07	30.52	**32.25**
	Conv-CRF	31.77	32.25	**34.16**

TABLE 3.2: Quantitative results for the SACAM-SegNet and single-branch models in percentage. The proposed network outperforms single-branch models.

ignore object boundaries and cover all adjacent pixels belonging to the same material category. A good example is the ceiling and the wall in the living room picture. Surprisingly, the SACAM-SegNet can even tell the difference between the scene outside the window and the scene drawing in the painting in the living room, and successfully classify them as glass and paper respectively. However, it is also noticed that the PAC-CRF refined SACAM-SegNet tends to predict wrong labels if the material region has rich textural clues. For example, the striped curtain covers the window in the kitchen. The PAC-CRF forces the network to label pixels between the stripes to different categories. This behaviour is not desired since it can give wrong boundary information. That is the reason why this chapter chooses to use a Conv-CRF refined model to generate the pseudo labels.

Table 3.3 compares the performance between SACAM-SegNet and BCAM-SegNet with the self-training approach. Without boundary loss, the SACAM-SegNet performs worse compared with the BCAM-SegNet. This shows that self-training alone does not result in the good performance of the BCAM-SegNet. The boundary information can stabilise the CAM-SegNet to learn from noisy pseudo labels and gradually correct the pseudo labels to achieve higher accuracy. The qualitative comparison can be found in Figure 3.5 and 3.6.

Models	SACAM-SegNet		BCAM-SegNet	
	Pixel Acc	mIoU	Pixel Acc	mIoU
Student 1	66.42	37.93	67.38	39.26
Student 2	67.26	38.97	68.18	39.81
Student 3	64.85	32.19	**69.27**	**40.98**

TABLE 3.3: Quantitative performance of the CAM-SegNet trained on augmented LMD with the self-training approach. The values are reported in percentages.

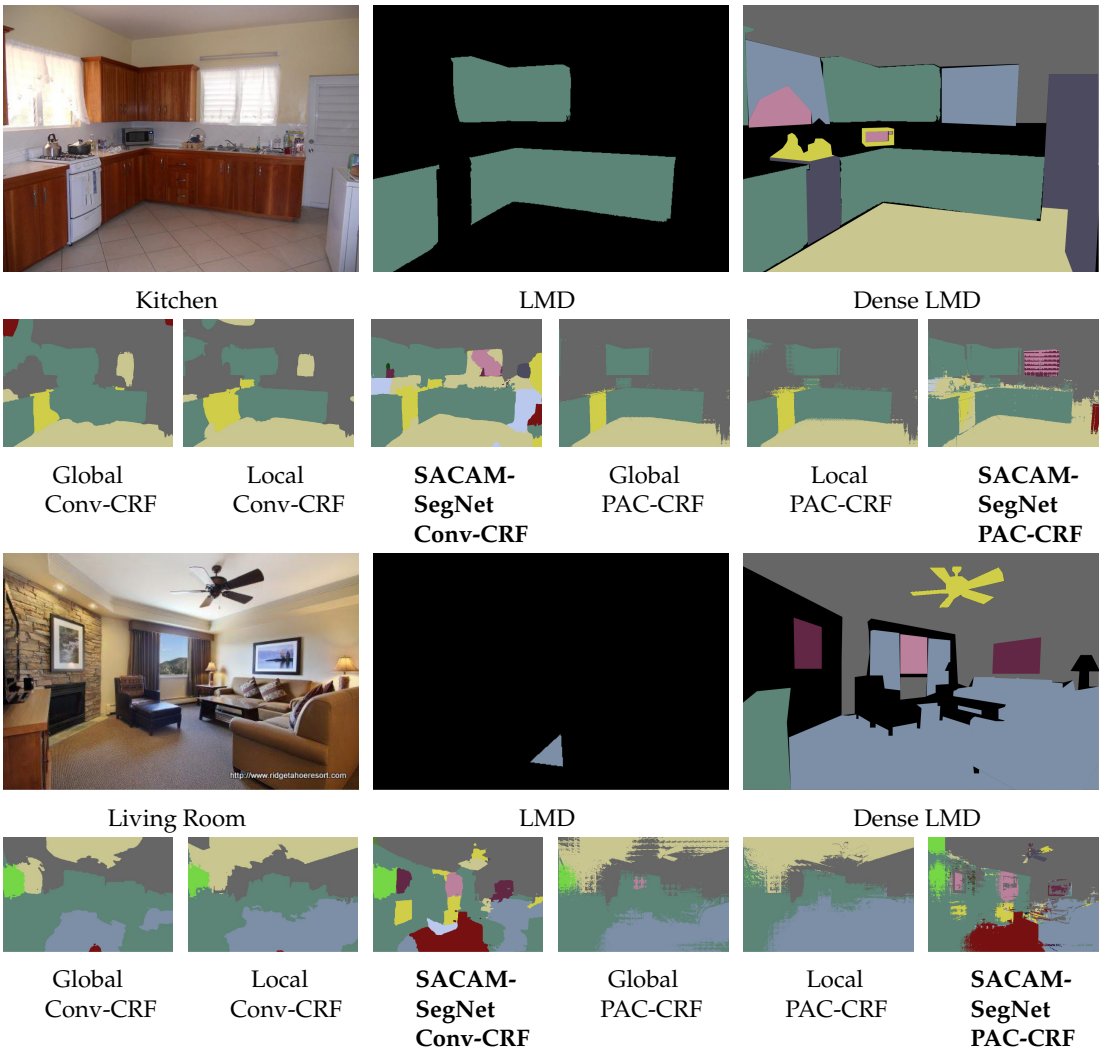


FIGURE 3.4: Dense material segmentation results for Kitchen image and Living Room image. The sparsely labelled images are taken from LMD, and densely labelled with all known material categories manually.

3.6 Conclusion

This chapter proposed a hybrid network architecture and training procedure to combine contextual features and material features. The effectiveness of the CAM-SegNet is validated with boundary contextual features. It is showed that the combined features can help the network to recognise materials at different scales and assign the pixels around the boundaries to the correct categories. However, the patch size is fixed when extracting the material features. This may not be the best choice for all the images since the areas that materials can cover vary within and across images. In consideration of this problem, in Chapter 4, the DBAT is proposed to aggregate features extracted from cross-resolution patches to improve the material segmentation performance.

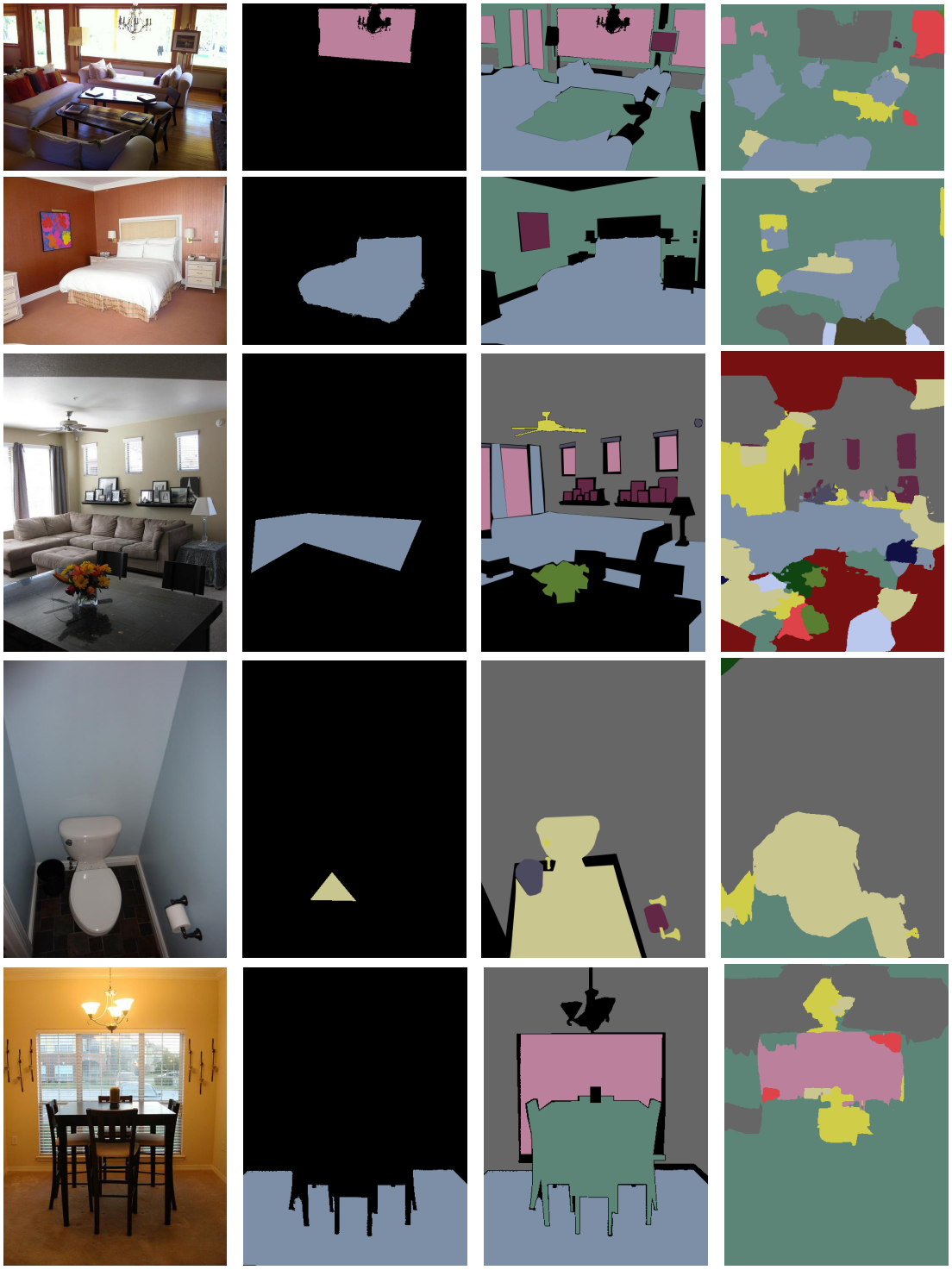


FIGURE 3.5: Dense material segmentation results for SACAM-SegNet, refined with Conv-CRF.

FIGURE 3.6: Dense material segmentation results for BCAM-SegNet, refined with Conv-CRF. The self-training approach is repeated three times.

Chapter 4

DBAT: Dynamic Backward Attention Transformer for Material Segmentation with Cross-Resolution Patches

This chapter introduces the contributions of my second-year PhD research, a single-branch network DBAT that can learn from cross-resolution image patches to make material predictions. This research is built on the first-year contribution, CAM-SegNet, with the goal of addressing its two main drawbacks: 1. the computational cost is doubled due to the use of multiple branches for learning both contextual and material features, and 2. the fixed patch resolution used for learning material features may not be optimal for all images. The structure of this chapter is similar to Chapter 3, which focuses on the methodology and experiment analysis.

4.1 Research Question and Motivation

This chapter aims to address the research question of how to effectively learn material features from cross-resolution image patches using a single-branch network architecture. The work in Chapter 3 and in (Schwartz and Nishino, 2020, 2016; Schwartz, 2018) indicate that the material features extracted from image patches can generalise the network to unseen images and achieve acceptable performance on material segmentation task. However, the resolution of the image patches is fixed, which may not be the best choice to extract material features. As affected by the camera working distance d_w and FoV, the areas that materials cover vary within and across images. Ideally, small patch

resolution should be applied to the boundary between materials, and large patch resolution can be used to cover as much information as possible for the region belonging to a single material.

4.2 Overview

Instead of searching for a fixed patch resolution, this chapter devises a simple yet effective transformer architecture, DBAT, to aggregate cross-resolution features. Inspired by the hierarchical architecture of Swin transformer (Liu et al., 2021b), which gradually merges image patches to get a global view, the DBA module is proposed in Section 4.3.1 to aggregate the intermediate features extracted from image patches with different resolutions. Concretely, a transformer feature map from a shallow layer contains features extracted from local patches (Raghu et al., 2021), especially when using window-based self-attention. The proposed DBAT merges adjacent patches at each transformer stage to enlarge the patch resolution, and aggregates multiple intermediate feature maps so that it can identify the materials with cross-resolution patch features. To cope with the flexibility of d_w and FoV, a set of pixel-wise attention masks, which represent the dependency on each patch resolution, are predicted in the DBA module to dynamically aggregate the feature maps. These masks are calculated from the deepest feature map (Map_4 in Figure 4.2) since it holds a relatively global perspective of the input image. Before feeding the aggregated feature into the decoder, Section 4.3.2 further proposes a feature merging module which ensures the aggregated feature can learn complimentary features through an attention-based residual connection.

The effectiveness of the proposed DBAT is examined through a comparison with well-known segmentation networks that can achieve real-time performance (at least 24 frames per second). The DBAT beats SOTA real-time models when evaluated on the sparsely labelled LMD (Schwartz and Nishino, 2020) and OpenSurface Database (Bell et al., 2013a). In particular, with modern optimisation strategies such as learning rate warm-up (Gotmare et al., 2018) and polynomial decay (Mishra and Sarawadekar, 2019), the DBAT reaches an average Pixel Acc of 86.85% on LMD, which is 21.21% higher than CAM-SegNet in Chapter 3, and outperforms the second-best model in this chapter by 2.15%.

4.3 Dynamic Backward Attention Transformer

This section explains the DBAT structure in detail. As shown in Figure 4.1, the DBAT consists of three modules: the backbone encoder, the dynamic backward attention module, and the feature merging module. The encoder is responsible for extracting

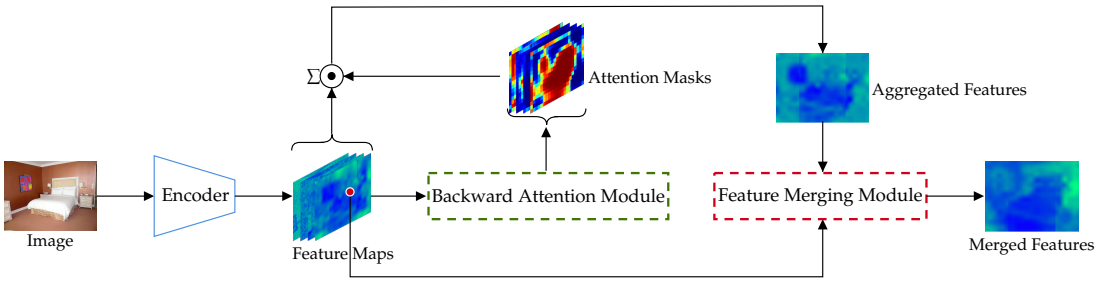


FIGURE 4.1: The architecture of the Dynamic Backward Attention Transformer. The symbol $\sum \odot$ represents the sum of element-wise production.

cross-resolution feature maps with window-based attention and patch merging. The proposed DBA module predicts per-pixel attention masks to aggregate the cross-resolution feature maps extracted from the encoder. The feature merging module guides the DBA module to extract features that are complementary to the last stage encoder output, which holds a global view of the image. Finally, the merged features, which have both enhanced cross-resolution features as well as global features, are passed into a segmentation decoder to make the material predictions.

4.3.1 Dynamic Backward Attention

The DBA module depends on a backbone encoder to extract feature maps from cross-resolution patches. There are multiple approaches to design the encoder. One possible choice is to employ multiple branches that learn from varying-sized patches, sharing the features during training, and tuning the number of trainable parameters with the patch size. Another option is to utilise non-overlapping Conv kernels (Yamanakkanavar and Lee, 2020) and enlarge the patch size through a pooling layer.

This research chooses the transformer as a suitable encoder to extract cross-resolution patch features, as it is inherently designed to process image patches (Dosovitskiy et al., 2020) and demonstrates promising results for vision tasks (Dosovitskiy et al., 2020; Liu et al., 2021b,a; Radford et al., 2021). Another reason is that, according to recent research (Raghu et al., 2021), the self-attention module can adapt its equivalent attention distance by varying the weights of each pixel. In this study, the equivalent attention distance is defined as the average Euclidean distance between two pixels, weighted by the attention weight. If the predictive accuracy on the training set improves by increasing the attention weights of neighbouring pixels, the network is considered to prefer local features. By allowing the network to choose from a large number of patch sizes through attention weights, the DBAT encoder can efficiently encode features at different resolutions, despite undergoing only four stages, which will be discussed in Chapter 5.

Figure 4.2 shows the way the DBA module aggregates the cross-resolution features. With the assumption that the features at each transformer stage can preserve spatial

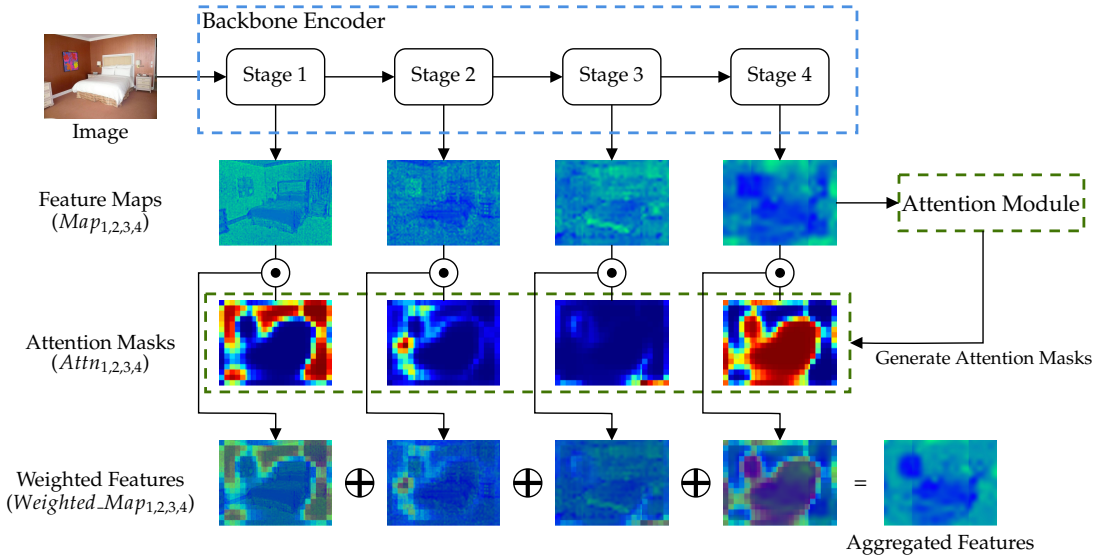


FIGURE 4.2: Structure of the DBA module. It performs a weighted sum across the feature maps, $Map_{1,2,3,4}$, to produce the aggregated feature. The weights are dynamically estimated based on the input image through the attention module, which takes the fourth feature map Map_4 as input. The sum $\oplus$ and product $\odot$ operations are performed element-wise.

location information (Raghu et al., 2021), this section proposes to aggregate these features through a weighted sum operation for each pixel of the feature map. For stage i, the feature map spatial size can be computed as $(\frac{H}{2\times 2^i}, \frac{W}{2\times 2^i})$, where H and W are the input image height and width. The attention weights $Attn_i$ are predicted from the last feature map, Map_4 with a 1×1 Conv layer. To perform the aggregation operation, it is necessary to normalise the attention weights with SoftMax operation so that the weights across the masks sum to 1 at each pixel location. It is worth noting that the spatial shapes of the feature maps should be the same in order to perform the pixel-wise product between Map_i and $Attn_i$. Moreover, the shapes of $Attn_i$ should be the same as well to normalise the per-pixel attention masks. This section sets $Attn_i$ to be the same size as Map_4, and downsamples $Map_{1,2,3}$ to the shape of Map_4 so that the computation and memory overhead can be minimised. The attention mechanism is expressed by Equation (4.1). Here the product $\odot$ and sum $\sum$ operations are all performed element-wise.

$$Aggregated\ Feature = \sum_{i=1}^{i=4} Attn_i \odot Map_i \tag{4.1}$$

4.3.2 Feature Merging Module

Although the DBA module aggregates cross-resolution features, it is not guaranteed that the aggregated features can improve network performance. Moreover, it is not

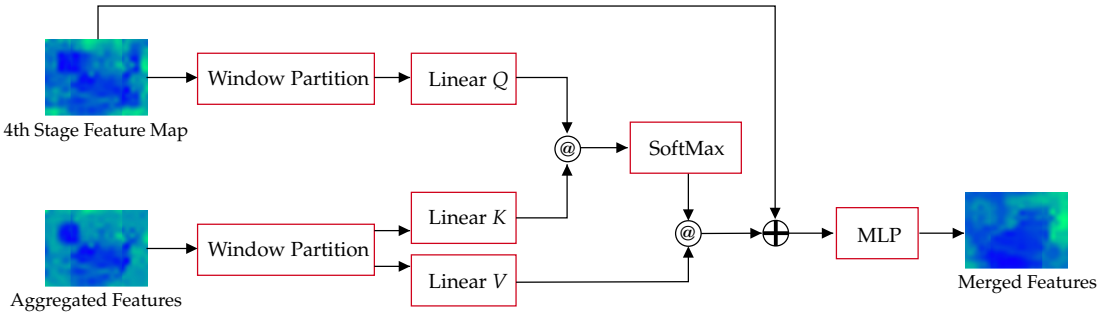


FIGURE 4.3: The feature merging module. It merges the relevant cross-resolution information from the aggregated patch feature into Map_4, through the window attention mechanism and residual connection.

desired to drop all global or semi-global features since they can limit the possible material categories in a given context (Schwartz and Nishino, 2016). Therefore, the feature merging module is proposed to guide the DBA module to enhance the local features without losing the original backbone features. This module consists of a global-to-local attention as well as a residual connection (He et al., 2016). The simple residual connection, *e.g.* $Merged\ Feature = Map_4 \oplus Aggregated\ Feature$, can ensure the DBA module learns complementary features. However, this simple addition operation would over-emphasise Map_4 and break the DBA module which aggregates features linearly. Therefore, this section chooses to bring in non-linear operations with the global-to-local attention (Shen et al., 2022; Liu et al., 2021b; Tu et al., 2021; Xu et al., 2019), which identifies the relevant information in the aggregated cross-resolution patch features through attention mechanism, and merges it into Map_4, as illustrated in Figure 4.3. The query matrix Q is predicted from Map_4 and it is applied to the key matrix K from the aggregated features. The matrix multiplication (represented by the @ symbol) between K^T and Q produces the attention alignment scores. Then the scores are normalised with SoftMax to extract relevant information from the value matrix V. Here the window-based attention in (Liu et al., 2021b) is used as well. With the DBA module and the feature merging module, the mechanism of DBAT can be described as enhancing the material-relate local features by injecting cross-resolution patch features into Map_4.

4.4 Experiment Configurations

This section gives a detailed explanation of the material segmentation datasets and network configurations. While Chapter 3 serves as the foundation for this chapter, the specifics diverge significantly. Notably, a new dataset named OpenSurfaces is utilised in this chapter, and the network training approach is different.

4.4.1 Material Segmentation Datasets

The present study evaluates the proposed DBAT using two datasets, namely LMD (Schwartz and Nishino, 2016, 2020; Heng et al., 2022b) and OpenSurfaces (Bell et al., 2013a). LMD comprises 5,845 low-resolution images acquired from indoor and outdoor sources, which have been manually labelled with 16 mutually exclusive material classes. On the other hand, OpenSurfaces contains 25,352 high-resolution indoor images labelled with 45 material categories. However, both datasets suffer from sparsely or coarsely labelled segments, as labelling images with material labels presents a significant challenge (Heng et al., 2022b; Schwartz and Nishino, 2016). One of the key difficulties faced by annotators is that materials are often treated as properties of objects (Bell et al., 2013a; Schwartz and Nishino, 2016). Consequently, material segments tend to be labelled within object boundaries, which is undesirable as the material region should be marked independently of its context. For instance, when annotating a scene depicting a wooden bed on a wooden floor, the wood segment should ignore the object boundary and cover all wood pixels. Additionally, OpenSurfaces is highly unbalanced, with only 27 out of the 45 material classes having more than 60 samples. Among all the samples, 39.44% are segmented as 'wood' or 'painted'. These limitations make the evaluation on OpenSurfaces less reliable compared to that on LMD. Therefore, this study mainly focuses on LMD, and the evaluation on OpenSurfaces will be presented as an additional piece of evidence. Notably, a recent dataset, MCubeS (Liang et al., 2022), has been proposed to perform material segmentation on outdoor images using multimodal data such as imaging with near-infrared and polarised light. However, since this study concentrates on material segmentation using indoor RGB images, the evaluation of MCubeS is not included in this chapter.

4.4.2 Evaluation metrics

In this section, the networks are evaluated with three metrics: Pixel Acc, Mean Acc, and mIoU. As discussed above, the material annotations may not cover the whole material region. As a consequence, the mIoU numerator would be much smaller than it should be. This situation is especially severe for LMD which was annotated sparsely on purpose (Schwartz and Nishino, 2016, 2020). Therefore, the mIoU is not reported for LMD. In addition to evaluating segmentation performance, this chapter reports the resources required for each model, including #params and #flops per forward propagation. To select model variants for evaluation, this chapter sets a selection criterion with the frames per second (FPS). The model variants that can support real-time inference (FPS larger than 24) are compared with the DBAT in this chapter.

4.4.3 Implementation details

The networks reported in Section 4.5 are pre-trained on ImageNet (Deng et al., 2009). This pre-training step is expected to teach the network with prior knowledge about contextual information such as scenes and objects. According to Schwartz and Nishino (2016), contextual information can reduce the uncertainty in material segmentation. Therefore, the network should learn material features more efficiently with a pre-training strategy. The details will be discussed in Chapter 5. For the Swin backbone, its implementation follows the original paper and the window size of the self-attention is set to 7. The decoder used in this chapter is the FPN Lin et al. (2017a) for the reason that it can recognise small material regions well, as shown in Chapter 3.

As for the training process, it is noticed that networks may not perform well on LMD in Chapter 3. One possible reason is the use of the Adam optimiser, which is proven to impair the generalisation ability of networks (Wilson et al., 2017). To boost the network performance to the SOTA level, three advanced technologies were adopted: the AdamW optimiser (Loshchilov and Hutter, 2018), the linear learning rate warm-up (Ma and Yarats, 2021), and the polynomial learning rate decay (Ge et al., 2019). This chapter uses the AdamW optimiser to train the networks with batch size 16, coefficients β_1 0.9, β_2 0.999, and weight decay coefficient 0.01. Further, the learning rate is warmed-up from 0 to 0.00006 with 1,500 training steps, and decreased polynomially. The networks are trained on LMD for 200 epochs and on OpenSurfaces for five days. Furthermore, as LMD images have various resolutions, the training images are resized first so that the minimum borders are equal to 512. The resized images are then cropped into 512×512 patches to use batch training (Mu, 2014) and facilitate batch normalisation (Santurkar et al., 2018).

4.5 Segmentation Performance Analysis

4.5.1 Quantitative Analysis

Table 4.1 reports the segmentation performance of the DBAT as well as five other models, ResNet-152 (He et al., 2016), ResNest-101 (Zhang et al., 2020a), EfficientNet-b5 (Tan and Le, 2019), Swin-t (Liu et al., 2021b), and CAM-SegNet from Chapter 3. Their heaviest variants that can serve real-time inference are evaluated apart from the CAM-SegNet. Although the CAM-SegNet does not meet the real-time selection criterion, it is the most recent architecture at the time for the RGB-based material segmentation task. It is noticed that its architecture is suitable for the DBA module since it has a dedicated local branch. Therefore, in this chapter, its local branch is equipped with the DBA

Datasets	LMD		OpenSurfaces			–		
Architecture	Pixel Acc (%)	Mean Acc (%)	Pixel Acc (%)	Mean Acc (%)	mIoU (%)	#params (M)	#flops (G)	FPS
ResNet-152	80.68 ± 0.11	73.87 ± 0.25	83.11 ± 0.68	63.13 ± 0.65	50.98 ± 1.12	60.75	70.27	31.35
ResNeSt-101	82.45 ± 0.20	75.31 ± 0.29	84.75 ± 0.57	65.76 ± 1.32	53.74 ± 1.06	48.84	63.39	25.57
EfficientNet-b5	83.17 ± 0.06	76.91 ± 0.06	84.64 ± 0.34	65.41 ± 0.44	53.79 ± 0.54	30.17	20.5	27.00
Swin-t	84.70 ± 0.26	79.06 ± 0.46	85.88 ± 0.27	$\mathbf{69.74 \pm 1.19}$	57.39 ± 0.54	29.52	34.25	33.94
CAM-SegNet-DBA	$\underline{86.12 \pm 0.15}$	$\underline{79.85 \pm 0.28}$	$\underline{86.00 \pm 0.64}$	$\underline{69.61 \pm 1.08}$	$\underline{57.52 \pm 1.44}$	68.58	60.83	17.79
DBAT	$\mathbf{86.85 \pm 0.08}$	$\mathbf{81.05 \pm 0.28}$	$\mathbf{86.43 \pm 0.02}$	69.18 ± 0.08	$\mathbf{57.57 \pm 0.06}$	56.03	41.23	27.44

TABLE 4.1: Segmentation performance on the LMD and the OpenSurfaces. The FPS is calculated by processing 1000 images with one NVIDIA 3060ti. The uncertainty evaluation is reported by training the networks five times. The best performance is shown in bold text and the second best is underlined.

module and its performance is reported as the CAM-SegNet-DBA [1]. The evaluations are reported across five independent runs. The metric differences are reported in the order (Pixel Acc/Mean Acc/mIoU) with the additive method.

As shown in Table 4.1, the DBAT achieves the best accuracy on LMD among all the real-time models in terms of Pixel Acc and Mean Acc. Specifically, the DBAT achieves +0.85%/+1.50% higher than the second-best model, CAM-SegNet-DBA. It is also +2.54% /+2.52% higher than its backbone encoder, Swin-t. As for the OpenSurfaces, the DBAT beats the chosen models on Pixel Acc and mIoU. Moreover, its performance is comparable to the multi-branch CAM-SegNet-DBA (+0.50%/-0.62%/+0.09%) with 9.65 more FPS and 19.6G fewer FLOPs. It is worth noting that compared with the performance reported in Chapter 3 of CAM-SegNet, the DBAT improves the Pixel Acc by 21.21%. Moreover, the per-category analysis in Table 4.2 shows that the DBA module improves the recognition of materials that usually have uniform appearances but varying shapes, such as paper, stone, fabric and wood. This indicates that the cross-resolution features successfully learn from distinguishable material features.

As shown in Figure 4.4, the DBAT can segment the materials consistently to achieve narrow uncertainty bounds, especially for the category foliage from LMD. Unlike the other five models, almost all the reported runs of the DBAT are within the upper and lower whiskers except for the category asphalt and metal. For the evaluations on OpenSurfaces shown in Figure 4.5, the uncertainty bounds of the proposed DBAT are much narrower compared with other models ($\pm$0.02 p.p. for Pixel Acc, $\pm$0.08 p.p. for Mean Acc and $\pm$ 0.06 p.p. for mIoU)[2]. This indicates that the DBAT is robust to the network initialisation and can learn from image patches effectively. The CKA similarity score, which is 0.9583 for the DBAT, is calculated and averaged for every two checkpoints of the five individually trained networks to support this deduction.

[1]The CAM-SegNet-DBA is implemented by replacing the original local branch by a combination of non-overlapping Conv kernels and MLP. The patch resolution is enlarged by concatenating features within the kernel size.

[2]Here p.p. stands for the percentage points.

Model	ResNet-152	ResNeSt-101	EfficientNet-B5	Swin-t	CAM-SegNet-DBA	DBAT
Asphalt (4.87)	88.66 ± 0.17	**94.35 ± 0.27**	82.17 ± 2.80	91.83 ± 1.09	89.87 ± 1.94	88.66 ± 0.72
Ceramic (2.95)	65.29 ± 3.19	62.86 ± 0.67	73.34 ± 0.42	**75.35 ± 0.42**	75.01 ± 0.64	68.31 ± 1.31
Concrete (4.73)	50.89 ± 1.67	60.53 ± 2.00	59.36 ± 2.98	57.42 ± 4.88	**69.20 ± 2.81**	66.90 ± 1.07
Fabric (10.96)	85.53 ± 0.22	86.420 ± 0.92	85.33 ± 0.20	88.71 ± 0.50	90.79 ± 0.43	**93.14 ± 0.16**
Foliage (15.43)	93.55 ± 0.33	91.25 ± 1.16	88.21 ± 0.32	**95.57 ± 0.45**	94.04 ± 0.79	95.35 ± 0.12
Food (10.03)	90.27 ± 0.22	94.96 ± 0.34	**95.84 ± 0.14**	92.51 ± 0.83	95.19 ± 0.24	93.27 ± 0.22
Glass (1.94)	72.58 ± 2.50	68.33 ± 0.34	77.83 ± 0.94	77.95 ± 0.99	**84.88 ± 1.11**	73.27 ± 0.67
Metal (6.17)	75.35 ± 0.94	80.66 ± 0.34	76.67 ± 0.28	81.54 ± 1.36	**81.83 ± 0.48**	79.99 ± 0.51
Paper (1.76)	64.52 ± 2.87	71.14 ± 1.99	**77.21 ± 0.13**	63.05 ± 1.90	66.48 ± 1.43	73.83 ± 0.67
Plaster (2.54)	68.01 ± 0.53	**78.76 ± 0.62**	73.11 ± 0.64	78.12 ± 1.90	72.37 ± 1.03	71.43 ± 0.71
Plastic (1.47)	34.87 ± 1.21	36.07 ± 3.42	39.59 ± 0.64	51.64 ± 1.31	**52.07 ± 2.28**	50.62 ± 1.45
Rubber (1.08)	77.08 ± 3.61	79.57 ± 1.62	69.73 ± 0.29	**83.48 ± 0.67**	81.63 ± 1.79	82.61 ± 1.01
Soil (8.22)	73.27 ± 1.63	73.15 ± 2.67	79.73 ± 0.55	76.89 ± 1.11	80.39 ± 1.73	**84.25 ± 0.50**
Stone (3.13)	69.66 ± 1.42	52.12 ± 0.93	70.07 ± 0.76	73.05 ± 1.92	60.73 ± 2.76	**86.94 ± 0.95**
Water (11.17)	95.49 ± 0.33	**97.54 ± 0.28**	95.30 ± 0.32	95.78 ± 0.70	94.95 ± 0.69	97.12 ± 0.10
Wood (13.54)	76.05 ± 1.08	76.71 ± 1.23	86.69 ± 0.24	82.03 ± 1.11	87.63 ± 0.98	**90.53 ± 0.37**
Pixel Acc	80.68 ± 0.11	82.45 ± 0.20	83.17 ± 0.06	84.71 ± 0.26	86.12 ± 0.15	**86.85 ± 0.08**
Mean Acc	73.87 ± 0.25	75.31 ± 0.29	76.91 ± 0.06	79.06 ± 0.46	79.85 ± 0.28	**81.05 ± 0.28**

TABLE 4.2: Per-category performance analysis in terms of Pixel Acc (%). The networks are trained five times to report the uncertainty. The metrics are reported in percentages. The number after the material category is the pixel coverage (in percentage) of each material in the dataset.

4.5.2 Qualitative Analysis

Figure 4.6 shows the predicted material segmentation for three images. In Figure 4.6 (a), ResNet-152 (He et al., 2016), ResNest-101 (Zhang et al., 2020a), Swin-t (Liu et al., 2021b), and the modified CAM-SegNet-DBA segment the bed as fabric, the floor as plaster. However, in this image, the floor appears to be covered with a carpet whose material is fabric. One possible explanation is that for the scene bedroom, the floor and wall are typically covered with plaster. These networks fail to make predictions based on material features, but rely on contextual information, so they tend to use plaster as a label for predictions. The proposed DBAT and the EfficientNet-b5 (Tan and Le, 2019) break the object boundary and segment part of the floor as fabric. Moreover, there are fewer noisy pixels in the DBAT segmented image when compared with the EfficientNet-b5. This indicates that with cross-resolution patch features, the DBAT can identify materials densely and precisely even if it is trained on a sparsely labelled dataset.

The segmented materials in Figure 4.6 (b) and Figure 4.6 (c) provide more evidence that the DBAT can segment the images well with features extracted from cross-resolution patches. In Figure 4.6 (b), the boundary between the wooden window frame and glass-made windows in DBAT segmented image is more adequate than the segments predicted by other networks. In Figure 4.6 (c), the segmented fabric aligns well with the ground truth with no noisy predictions. Considering that the training of DBAT takes sparsely labelled segments, it is reasonable to say that DBAT learns the difference between materials from cross-solution patches.

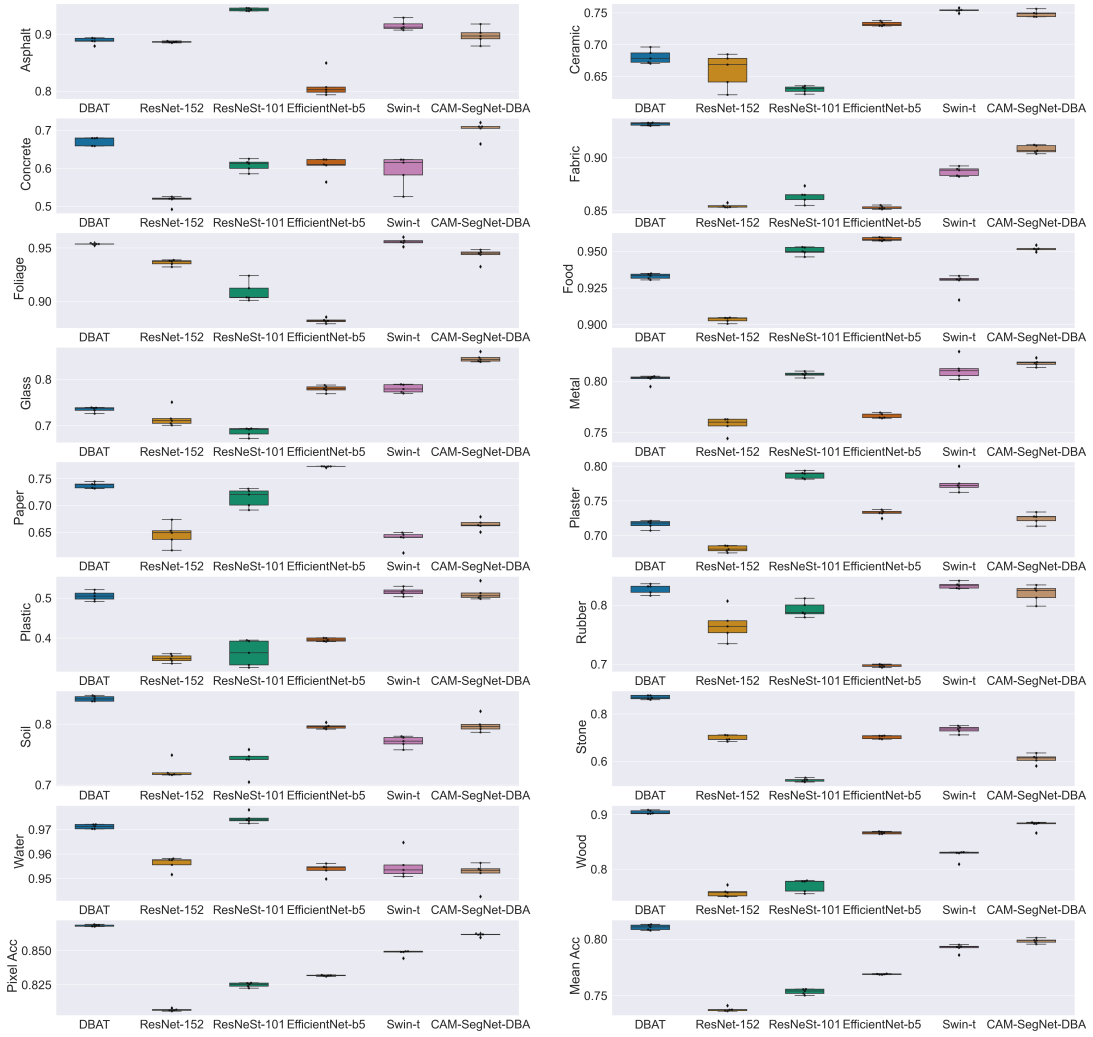


FIGURE 4.4: Boxplot of the performance on the LMD across five runs.

4.5.3 Ablation Study

This section studies the effectiveness of each component of the DBAT. In Table 4.3, the performance is reported after removing the feature merging module and the DBA module in sequence. Without the feature merging module, the Pixel Acc and Mean Acc decrease by 1.61 p.p. and 2.04 p.p. respectively. This shows the importance of the attention-based residual connection in improving performance. The performance drops by another 0.72 p.p. in Pixel Acc and 0.13 p.p. in Mean Acc after removing the DBA module. This shows that the DBA module that learns complementary cross-resolution features guided by the feature merging module can improve performance effectively.

In order to justify the network design, this section further studies the alternative implementations of the DBAT from three aspects: 1) how the attention masks are predicted; 2) how the feature maps are downsampled; 3) how aggregated features are merged with Map_4. The performance differences in Table 4.4 are reported by switching one of the

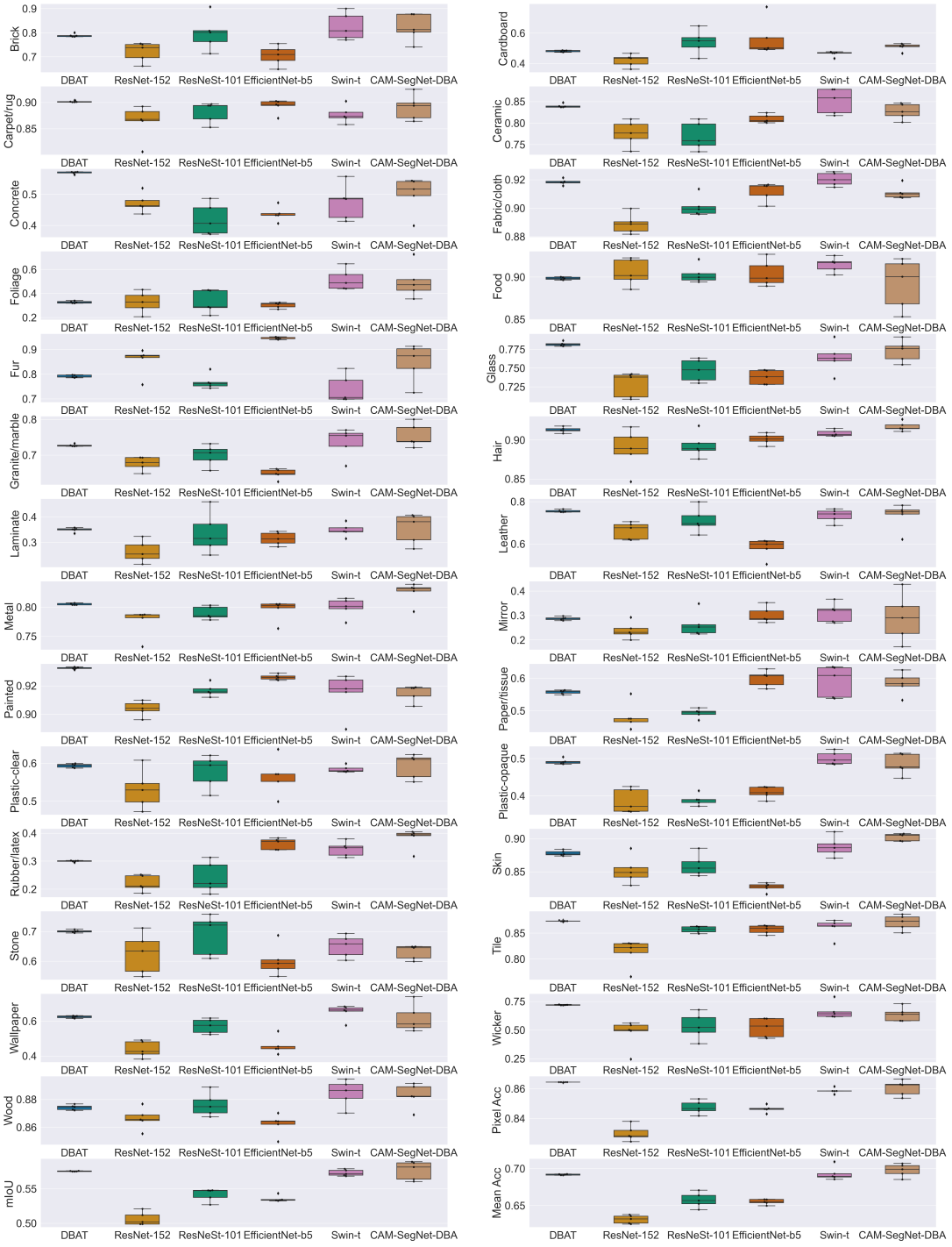
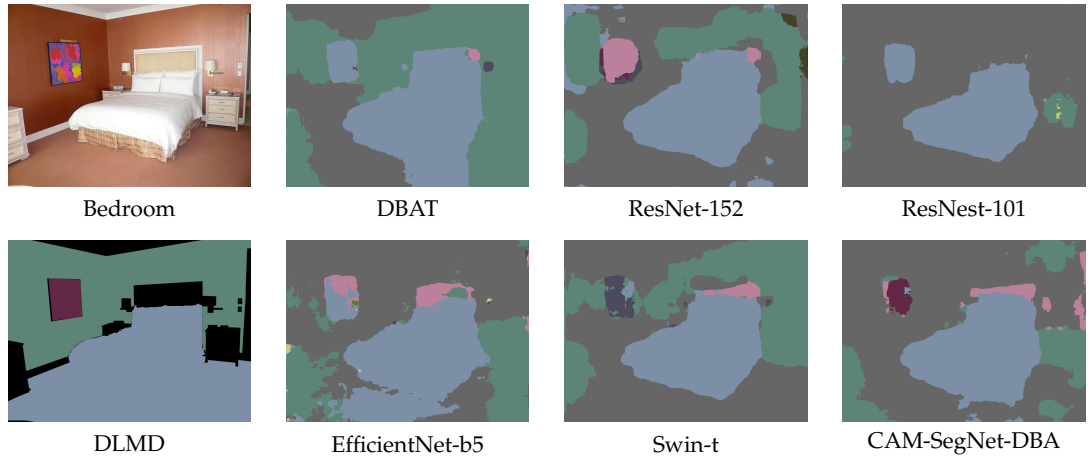
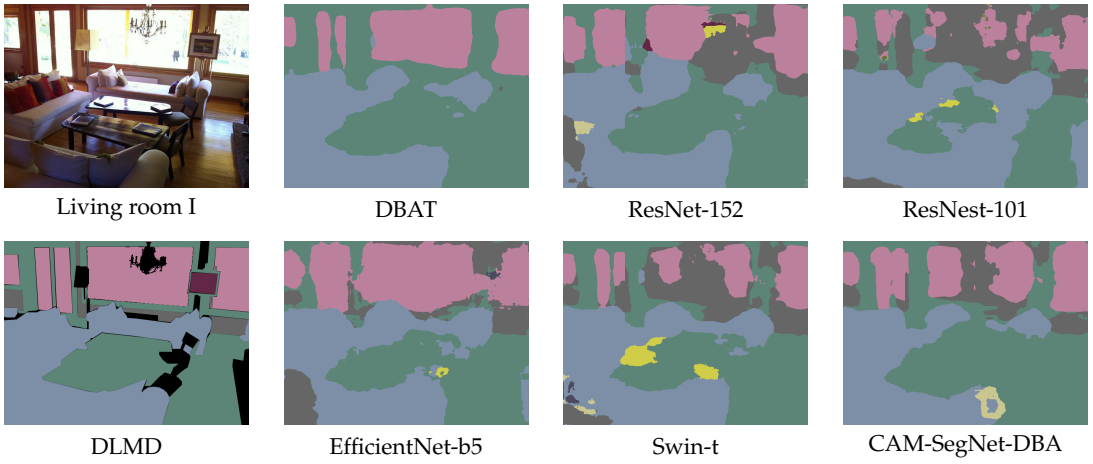


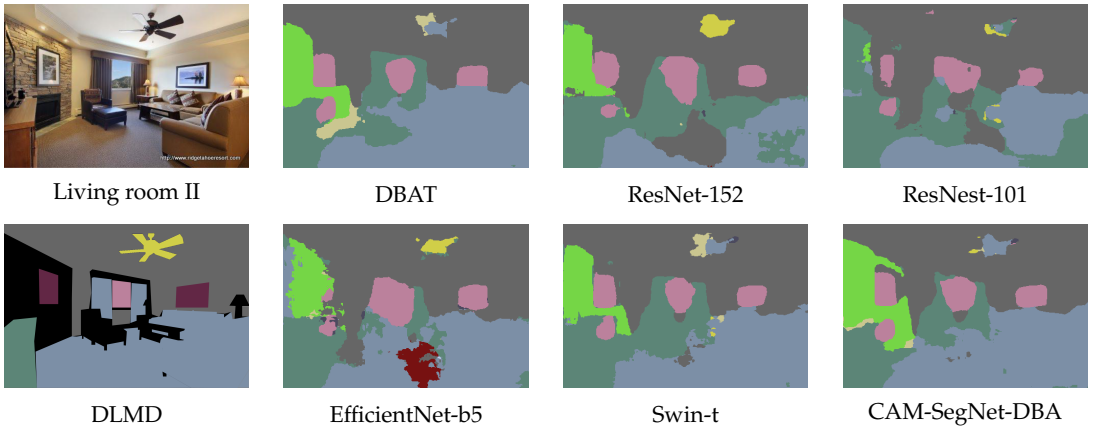
FIGURE 4.5: Boxplot of the performance on the OpenSurfaces across five runs.



(a) Segmented Images for Bedroom



(b) Segmented Images for Living room I



(c) Segmented Images for Living room II

FIGURE 4.6: Predicted segmentation of three scenes.

Architecture	Δ Pixel Acc	Δ Mean Acc
- Feature merging	-1.61	-2.04
- Dynamic backward attention	-2.33	-2.17

TABLE 4.3: The ablation study to analyse each component of the DBAT. The performance difference is reported in percentage points.

implementations in DBAT to its alternatives. As stated in Section 4.3.1, the proposed DBAT adopts convolutional kernels to generate the per-pixel attention masks (Chen et al., 2020b). By replacing the kernels to dilated ones (Wei et al., 2018), the receptive field is enlarged when predicting the masks. However, the DBAT performance decreases significantly by -2.15 p.p./-2.67 p.p.. This indicates that the local information is critical for the dynamic attention module to work well. Section 4.3.1 describes that the cross-resolution feature maps need to be downsampled so that the fixed-size attention masks can be applied. Originally DBAT uses the MLP to downsample the feature map. Instead of using such a trainable method, the network can also use a superficial non-parametric pooling layer, which decreases the performance by -0.88 p.p./-1.58 p.p.. The slight drop in Pixel Acc and the significant drop in Mean Acc suggest that the trainable downsampling method can help balance the performance of different material categories. As for the feature merging module, a simple residual connection slightly reduces the performance by -0.58 p.p./-0.64 p.p.. This highlights that DBAT needs a residual connection to guide the aggregation of cross-resolution features and the representation capability of the feature merging module may not be vital.

Implementation Choices		Δ Pixel Acc	Δ Mean Acc
Generate Attention Masks	CNN –> Dilated CNN	-2.15	-2.67
Downsample	MLP –> Average Pooling	-0.88	-1.58
Feature Merging	Attention –> Residual Connection	-0.58	-0.64

TABLE 4.4: The study of implementation choices in each component of the DBAT. The performance difference is reported in percentage points.

4.6 Conclusion

This chapter presented a novel single-branch network that dynamically enhances material features by aggregating cross-resolution patch features. The proposed DBAT outperforms selected real-time models on two datasets and achieves comparable performance with lower computational cost compared to the multi-branch CAM-SegNet. While the network performance has been improved by learning from image patches following the patch learning assumption (Schwartz and Nishino, 2020), it is still unclear

what features the network learns to make material predictions since understanding the network behaviour using human-understandable concepts remains a challenging task. Rather than focusing solely on improving material segmentation performance, the next chapter will investigate the interpretation of material features learned by the network. This will be accomplished by comparing them with features extracted from different tasks, such as object segmentation, and visualising attention masks, CKA heatmaps, and using network dissection methods.

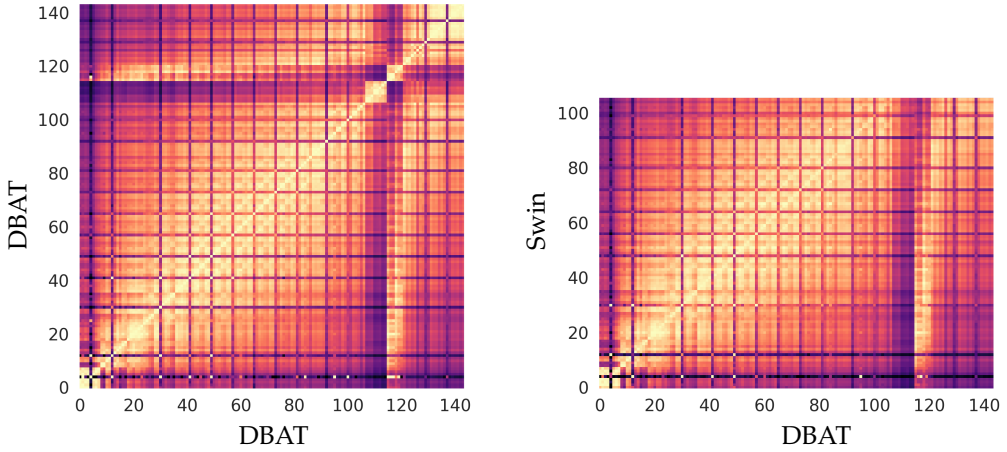
Chapter 5

Network Behaviour Analysis and Feature Interpretability of the DBAT

Similar to other network-based methods, the DBAT faces challenges in terms of interpretability. Ascertaining whether the network genuinely acquires material features through numerical evaluation or segmentation visualisation is a complex task. Therefore, this chapter further endeavours to interpret the network behaviour of the DBAT using statistical and visual tools, such as calculating the attention head equivalent patch size, visualising attention masks, and assessing the CKA heatmap (Nguyen et al., 2020; Raghu et al., 2021). In order to interpret the features with human-readable concepts, this chapter also employs the network dissection method (Zhou et al., 2018; Bau et al., 2017, 2019, 2020) to identify the features learned by the network by aligning layer neurons with semantic concepts. By analysing the semantic concepts of the extracted features, this chapter illustrates that the DBAT excels in extracting material-related features, such as texture, which is an essential property for distinguishing between various materials. By comparing the semantic concepts of features extracted by other networks trained with either material or object datasets, the results also indicate that the network architecture can influence the extracted features, and the patch-based design is indeed effective in compelling the networks to segment images based on material features.

5.1 Research Question

This chapter investigates the research question of how networks learn to make material predictions by studying the features they extract. The research question is centered on the process of feature learning in material segmentation tasks.



(a) The CKA matrix of the DBAT (b) The CKA matrix between DBAT and Swin

FIGURE 5.1: The CKA matrix where each position measures the similarity between the features extracted by two arbitrary layers. The brighter the colour is, the more similar features these two layers extract.

5.2 CKA Heatmap Analysis

This section teases apart the network behaviour through the CKA matrix (Nguyen et al., 2020; Raghu et al., 2021). Figure 5.1 visualises the CKA heatmap of the DBAT in (a) and the cross-model heatmap between DBAT and Swin in (b). The layers are indexed by the forward propagation order. Before layer 106, the DBAT shares the same network architecture as the Swin. The bright line in (b) connecting (0, 0) and (105, 105) gets darker when approaching point (105, 105). This indicates that the Swin backbone in the DBAT extracts similar features to when used alone at shallow layers and gradually learns something new when approaching deeper layers. The dark region in (a) from layer 106 to 113 reflects the attention masks predicted by the DBA module. By gathering the cross-resolution feature maps, the aggregated features contain information from both shallow and deep layers, illustrated by the bright region between layer 113 and 124. After layer 124, the feature merging module combines the relevant information from the aggregated features into Map_4, which is extracted from layer 106. This module produces a feature map that differs from the feature extracted by Swin, as shown by the points around (140, 100) in (b).

5.3 Attention Analysis

This section analyses the dynamic attention module by visualising the attention masks and calculating their descriptive statistics, including the average attention weights as well as the equivalent attention patch resolutions. Figure 5.2 shows the attention masks for images in the LMD test set. From $Attn_1$ to $Attn_4$, the patch resolution increases. It is discovered that the material covering multiple small objects or a small area tends to

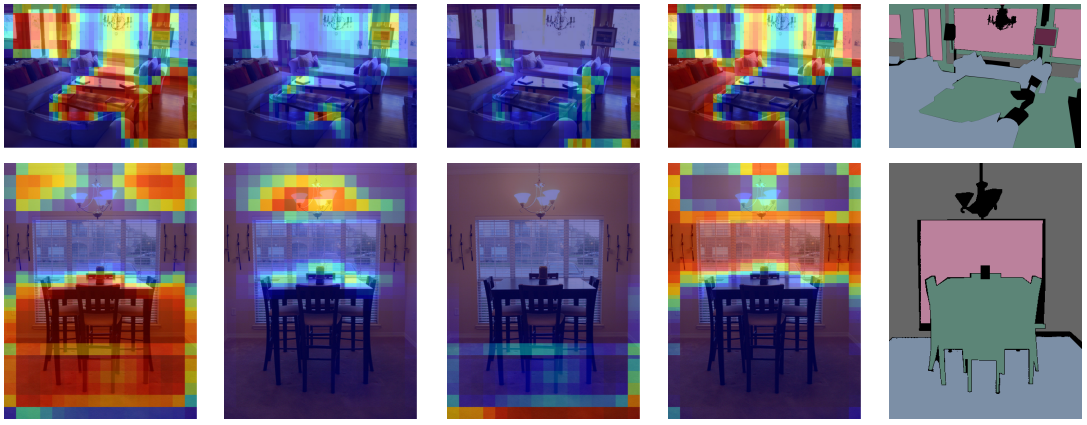


FIGURE 5.2: Attention mask visualisation. GT: ground truth. The densely labelled ground truth images are collected from DLMD in Chapter 3.

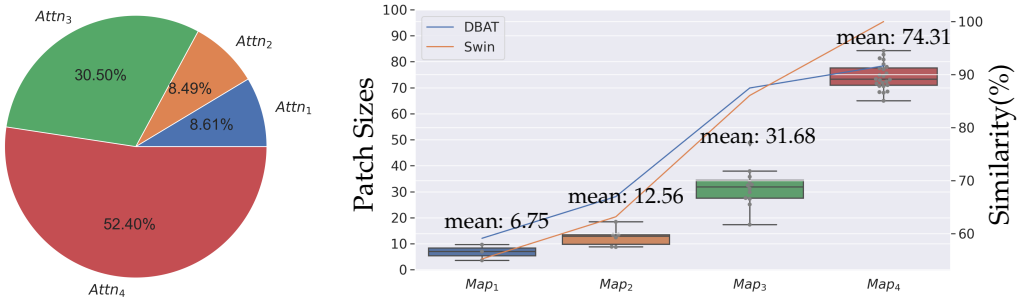
depend on the features extracted from small patch resolution. For example, the first column images in Figure 5.2 highlight the regions that the network concentrates on. The wooden area in the first row covers both the desk and the floor. The wooden chairs and the fabric floor are mutually overlapping in the second row. These objects or material regions are mutually overlapping, and small patches can isolate them and learn features at the boundaries.

Figure 5.3 shows the average attention weight in (a) and the equivalent patch size in (b, the box plot). The equivalent patch size, which represents the attention distance of each attention head, is calculated by transforming the attention diagonal distance in (Raghu et al., 2021) to the side length of a square. For a single pixel in the feature map at position i, j, its equivalent patch size can be calculated as:

$$\text{patch size}_{i,j} = \sqrt{\frac{1}{2HW} \sum_{k=0}^{k=H} \sum_{l=0}^{l=W} SoftMax(K^T Q)_{k,l} \times \sqrt{(k-i)^2 + (l-j)^2}} \tag{5.1}$$

where H, W are the height and width of the feature map, and $SoftMax(K^T Q)_{k,l}$ is the dependency from the self-attention module for pixel i, j on pixel k, l. As expected, the aggregated features mostly (52.40%) depend on $Attn_4$, which is thoroughly processed by the whole backbone encoder, with an average patch size of 74.31. Apart from $Attn_4$, the aggregated features also depend on feature maps extracted from small patch sizes. For example, $Attn_3$ is extracted from an average patch size of 31.68, and it contributes 30.50% to the aggregated features. Although $Attn_1$ is gathered from a shallow stage of the network, the aggregated features still depend on it to handle the overlapping material regions with a patch size of 6.75 on average.

To further illustrate the effect of the dynamic backward attention module, the similarity scores comparing one layer with $Map_{1,2,3,4}$ from the CKA matrix is reported in Figure 5.3. (b, the line plot). The blue line compares Map_4 from the Swin encoder, and the



(a) Averaged Attention Weights (b) Equivalent Attention Patch Sizes

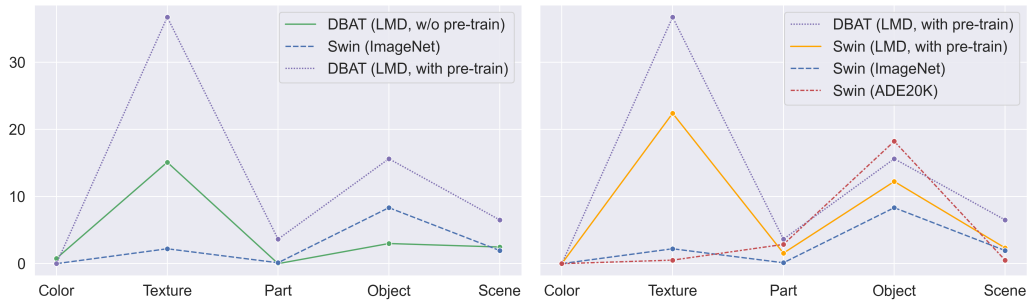
FIGURE 5.3: The descriptive analysis of attention weights.

orange line compares the aggregated features from the DBAT. The increased similarity scores against $Map_{1,2,3}$ clearly show that the aggregated feature depends on information from shallow layers.

5.4 Network Dissection

The network dissection method aligns the disentangled neurons of one network layer to semantic concepts (Bau et al., 2017, 2019). By counting the portion of neurons aligned to each concept, it is possible to give an understanding of what features the network learns. This chapter studies local concepts such as colour, texture and part, as well as global/semi-global concepts like object and scene. The neurons of the last encoder layer are selected to be analysed since they are more interpretable than shallow layers (Bau et al., 2017, 2019). Figure 5.4 (a) depicts how pre-training influences the features that the DBAT learns. Without pre-training, DBAT (the green line in Figure 5.4) has shown a preference for texture features, and a portion of its neurons can detect object and scene features. As shown by the dotted blue line, the Swin backbone (Liu et al., 2021b) trained with ImageNet (Martínez-Domingo et al., 2017) tends to detect object features. The DBAT has more neurons aligned to texture and object features when trained with a pre-trained backbone, shown as the purple line. The observations indicate two aspects: (1) the DBAT relies on texture features to solve the material segmentation task. (2) the pre-trained object detectors reduce the uncertainty in identifying materials and ease the training of texture detectors.

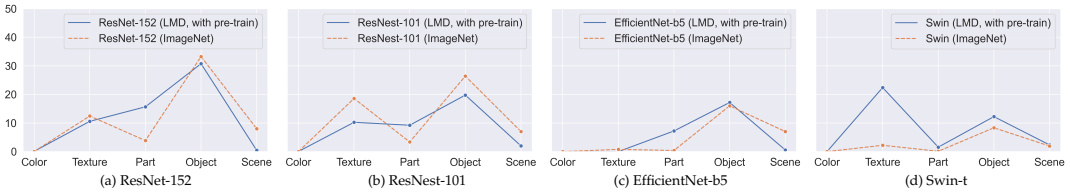
This chapter further analyses the feature difference between material and object tasks. In particular, Swin (Liu et al., 2021b) is trained with two object-related datasets: the ImageNet (Deng et al., 2009), an object classification dataset, and the ADE20K (Zhou et al., 2017, 2019a), an object-level segmentation dataset. As shown in Figure 5.4 (b), the significant difference is the lack of texture features in object-related tasks. This discovery highlights that enhancing features hidden in patches is a valid heuristic to improve the network performance on the material segmentation task.



(a) Concept alignments of DBAT with/without pre-training

(b) Concept alignments for material and object tasks

FIGURE 5.4: The analysis of the training process of the DBAT by counting the percentage of disentangled neurons aligned to each semantic concept.



(a) ResNet-152 (b) ResNest-101 (c) EfficientNet-b5 (d) Swin-t

FIGURE 5.5: The comparison between networks trained for object and material tasks, in terms of the percentage of neurons aligned to each semantic concepts.

Figure 5.5 dissects three more networks: ResNet-152 (He et al., 2016), ResNest-101 (Zhang et al., 2020a), EfficientNet-b5 (Tan and Le, 2019) on both material and object tasks. An interesting discovery is that although these networks achieve comparable performance, the features that they learn are different. For example, the ResNet-152 relies on texture features on both tasks and it learns more part-related features on material task compared with object task. Although three of the networks in Figure 5.5 learn texture features on the material task, a special exception is the EfficientNet-b5, which knows almost nothing about texture for both tasks. This phenomenon goes against the intuition that networks targeting material segmentation should learn texture features well since texture describes the appearances of materials. One reasonable explanation is that assigning material labels to object or object parts can cover the labelled material region and achieves a high accuracy since these material datasets are sparsely labelled. Therefore, this study calls for densely labelled material segmentation datasets for reliable evaluation and analysis.

The network dissection method provides a concise summary of the portion of neurons aligned to each concept. In addition to these numbers, it would be interpretable to visualise the regions where the filters are activated. In Figure 5.6, one example of the activated filters for the material fabric is visualised. The activation condition is introduced in Section 2.6.2. The blobs covering less than 200 pixels have been excluded for clarity. Remarkably, the filter exhibits activation across distinct objects featuring the

(A) Activated Regions (B) Overlaid Regions (C) Ground Truth

FIGURE 5.6: The activated regions (shown as white) of one filter.

same material. This suggests that the network has acquired information that is not dependent on specific shapes or instances, highlighting its ability to learn shape- and instance-agnostic features.

Chapter 6

MatSpectNet: Material Segmentation Network with Domain-Aware and Physically-Constrained Hyperspectral Reconstruction

After demonstrating in Chapter 4 that learning features from cross-resolution patches can enhance network performance, and in Chapter 5 that patch training strategy (Schwartz and Nishino, 2020) can improve the network ability to learn material-related features, it became evident that controlling the features that a network implicitly learns for making predictions is not trivial. In order to learn material features, it is necessary to make strong and proper assumptions about the materials involved to design the networks. However, although the network improves the segmentation performance, these assumptions may not be confirmed as valid unless they are interpreted by semantic labels, which is also limited to the number of semantic concepts available. Therefore, during the third year of my PhD study, the research focus shifted towards explicitly learning material features using material property measurement methods. The proposed MatSpectNet leverages hyperspectral reconstruction method to recover the corresponding hyperspectral images from RGB images, which explicitly measures the optical properties of materials.

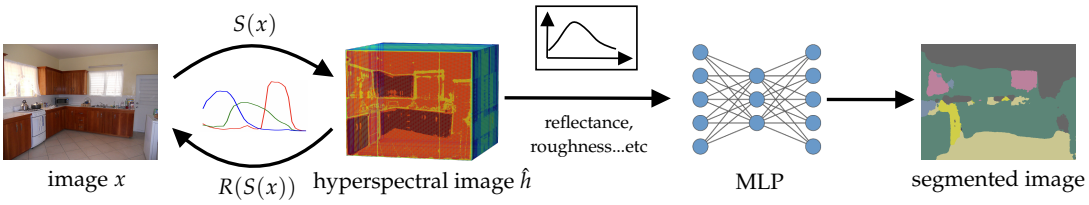
6.1 Research Question and Motivation

The focus of this chapter is to explore the research question of how to learn material features explicitly from hyperspectral images in situations where only RGB images are available. Although recent studies indicate that it is possible to achieve acceptable performance with annotated RGB datasets (Heng et al., 2022b,a; Schwartz, 2018; Schwartz and Nishino, 2020; Bell et al., 2015a), the experiments in (Liang et al., 2022; Mao et al., 2022) show that additional measurements of light such as near infra-red and laser beam reflection can distinguish materials more robustly. The theory is that the spectral profile of reflected electromagnetic waves is unique to various materials (Saragadam and Sankaranarayanan, 2020; Lichtman and Conchello, 2005; Colthup et al., 1990). Since spectral cameras (Behmann et al., 2018) can capture the spectral profile of surface materials, it is feasible to use the hyperspectral images they produce for material segmentation.

While hyperspectral imaging has been widely used in geoscience and remote sensing (Zhong et al., 2016; Kalman and Bassett III, 1997; Li et al., 2022b; Xue et al., 2021; Mehta et al., 2021; Liu et al., 2019a) over twenty years, the cost of collecting hyperspectral images hinders its widespread adoption in material segmentation for daily scenes (Stuart et al., 2022). A spectral camera can take a long acquisition time to scan a megapixel hyperspectral image with sufficient signal-noise ratio since the same amount of light has to be sampled at hundreds of wavelength bands (Behmann et al., 2018; Zhang et al., 2019a). This problem necessitates concessions in image spatial and spectral resolution. In addition, the ambient light should be able to cover the entire operating spectrum range, so the spectral camera should be used under daylight or halogen-based illumination. Before taking the hyperspectral images, the camera has to be calibrated with the measurement of black and white reference samples to analyse the material reliably (Behmann et al., 2018; Shaikh et al., 2021). The stringent lighting requirements further restrict the application of hyperspectral images in indoor and motion scenes.

6.2 Overview

In order to make spectral information more accessible for computer vision applications, researchers have been working on recovering spectral information from more easily obtainable data sources, such as RGB images (Arad et al., 2022). Over the past three years, several methods (Li et al., 2020; Hu et al., 2022; Cai et al., 2022c) have successfully improved the accuracy of reconstructed hyperspectral images. However, it remains unclear how these methods generalise to images captured by different camera models, as this aspect has not been explicitly investigated. In consideration of this problem, this chapter proposes a novel MatSpectNet to enhance the quality of recovered hyperspectral images on material datasets lacking RGB-hyperspectral image pairs. Figure



(a) physically-constrained spectral recovery (b) hyperspectral image processing

FIGURE 6.1: The overarching structure of the proposed MatSpectNet. It comprises two primary components: (a) Recovering the hyperspectral image from the RGB image by leveraging the physical camera model, and (b) Extracting material characteristics from the spectral channel through a combination of trainable filters and observations.

6.1 shows that the proposed MatSpectNet consists of two main sections. The network first learns to recover the hyperspectral images with the physical model of the camera, which serves as a constraint to ensure that the hyperspectral images preserve their physical property (Section 6.3.1, 6.3.2, illustrated in Figure 6.1, (a)). Then the recovered hyperspectral images are processed using interpretable spectral filters followed by a MLP and combined with other observations such as roughness to improve the segmentation quality (Section 6.3.3, 6.4, shown in Figure 6.1, (b)). The corresponding segmentation decoder will be introduced in Section 6.5.4.

6.3 Material Hyperspectral Network

This section introduces the components of the proposed MatSpectNet. Similar to the CAM-SegNet in Chapter 3, the training procedure of MatSpectNet is complicated and requires more space to introduce the loss terms and understand the network. Therefore, the training strategies are also included in this section.

6.3.1 Physically-Constrained Spectral Recovery

The spectral recovery network learns the transformation from an sRGB image $x \in \mathcal{X}$ to its corresponding hyperspectral image $\hat{h} \in \mathcal{H}$ via $S : \mathcal{X} \rightarrow \mathcal{H}$ (Cai et al., 2022c; Li et al., 2022a; Agarla et al., 2022; Arad et al., 2022). However, evaluating the quality of reconstructed hyperspectral images is challenging due to the absence of measured hyperspectral images in material datasets. To address this issue, this section introduces an RGB transformation network $R : \mathcal{H} \rightarrow \mathcal{X}$ that maps the reconstructed hyperspectral image $\hat{h}$ back to the corresponding sRGB image, as shown in Figure 6.2. The best parameters $\theta^* = \arg\min_\theta L_{trans}$ are selected by monitoring the loss term for the material datasets:

$$L_{trans} = L_{MSE}(x, R(S_\theta(x))) \tag{6.1}$$

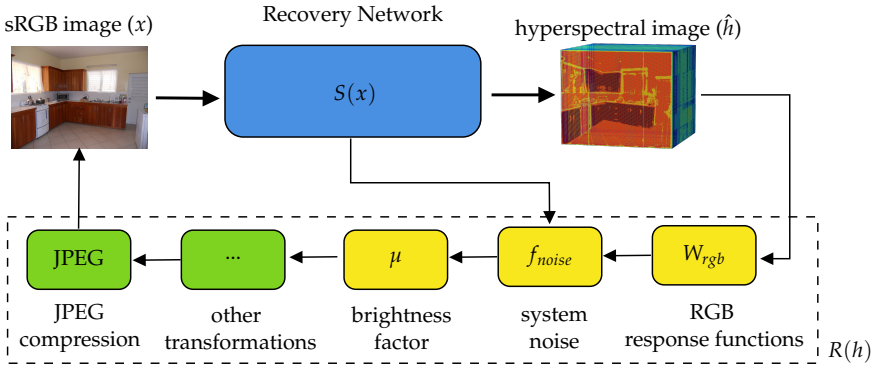


FIGURE 6.2: The learnable components in the RGB transformation network, $R(h)$, which models the physical camera, simplified from (Can Karaimer et al., 2019). The yellow components model the camera model with explicit equations, and the green components are modelled by network components. In a real camera, noise reduction happens before the in-camera transformation happens. In this thesis, the system noise is controlled to a low level manually to omit the noise reduction process.

To recover accurate hyperspectral images for material datasets by minimizing the transformation loss, the design of $R(h)$ must reflect the physical relationship of the RGB and hyperspectral image pair. Hence, a simple network, as used in previous works such as (Mehta et al., 2021), is not sufficient. To address this, $R(h)$ explicitly incorporates the physical RGB camera model including response functions, brightness, and system noise, as shown in Equation 6.2:

$$R(h) = f_{jpeg}(\mu f_{noise}(W_{rgb}h)) \tag{6.2}$$

where the hyperspectral image $h \in \mathbb{R}^{n\_bands \times H \times W}$ and the RGB response functions are formatted as a matrix $W_{rgb} \in \mathbb{R}^{3 \times n\_bands}$. The camera noise is included by the function f_{noise}, and μ is the brightness factor. Here, n_bands is the number of spectra bands sampled by the response functions. The function f_{jpeg} models the in-camera processing and compression noise introduced by the Joint Photographic Experts Group (JPEG) compression algorithm. All components are trainable in Figure 6.2 and will be explained in the following paragraphs.

6.3.1.1 RGB Response Functions

All digital light sensors exhibit varying sensitivity to different wavelength ranges of light through spectral response functions (Tropp, 2017). Specifically, trichromatic cameras or three-colour image sensors, inspired by human colour perception, have unique spectral response functions in their red (R), green (G), and blue (B) channels based on the tristimulus theory (Smithson, 2005). That is to say, knowing the measurement of the spectral reflection h and the response matrix W_{rgb}, sampled at the same bands, the noiseless RGB image rgb_{clean} can be obtained as $W_{rgb}h$, on the condition that h is properly calibrated (Ji et al., 2021; Behmann et al., 2018). However, the RGB values are not

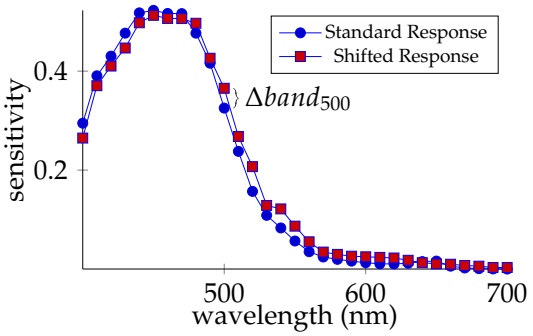


FIGURE 6.3: The illustration of the predicted band shift for the blue channel response curve.

consistent across various camera models, even under the same capturing conditions. This is because each model of three-colour image sensors responds differently to light due to their unique RGB spectral response functions. Hence, the RGB values are device-specific and not interchangeable among different camera models.

In order to handle images taken by different camera models (Schwartz and Nishino, 2020; Bell et al., 2015a, 2013a), this section proposes to learn the sensitivity displacement $\Delta band_i$ at each spectral band i compared with standard response functions based on the input sRGB image, as depicted in Figure 6.3. The standard response curves are sourced from the ARAD_1K dataset, which is based on physical measurements of a Basler Ace 2 camera (Arad et al., 2022). To learn the sensitivity displacement, the spectral recovery network $S(x)$, which obeys the encoder-decoder architecture, is re-designed by attaching one auxiliary path to the encoder. As shown in Figure 6.4, the encoder output is processed with the repetitive spectra processing module comprising 1×1 convolutional kernels and average pooling. The 1×1 convolutional kernel is responsible to learn from the channel information since the band difference is applied to each pixel individually. The average pooling downsamples the feature map to correct the prediction. Moreover, the response curves should learn from long, middle, and short wavelength regions for R, G, and B channels respectively. To keep maintain the functionality of the response curves, the band differences are aggregated as a loss term L_{band}:

$$L_{band} = \sum_{r,g,b} \sum_{\lambda} band_{\lambda|r,g,b} \times ||\Delta band_{\lambda|r,g,b}|| \tag{6.3}$$

where $band_\lambda$ is sampled from the standard response curve at wavelength λ for channel R or G or B. In this way, the highly sensitive region incurs severe penalties, causing its displacement $\Delta band_\lambda$ to be zero and preserving its functionality. In the experiment, the MST++ (Cai et al., 2022c) is chosen to be the spectral recovery network $S(x)$ and the spectra processing module in Figure 6.4 is repeated three times with channel numbers 124, 62, 31+3 where 31 of them is the number of spectra band sampled between 400nm and 700nm with step 10nm, and the other 3 scalars are the noise parameters and brightness factor explained in following sections.

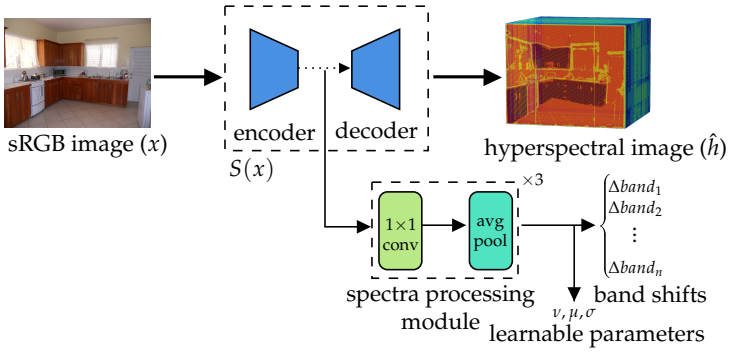


FIGURE 6.4: The network architecture to predict the learnable parameters and band shifts.

6.3.1.2 Camera System Noise and Brightness

Since the hyperspectral image h in the ARAD_1K datasaet is calibrated with white and dark reference samples to measure the actual reflectance rather than signal intensity, the recovered hyperspectral image $\hat{h}$ is projected into noiseless RGB rgb_{clean} by $W_{rgb}\hat{h}$. However, for realistic cameras, the camera system noise caused by unwanted variations in the signals produced by the image sensor and processing circuitry can reduce the image quality, particularly in low light conditions or high ISO settings (Baxter and Murray, 2012; Shin et al., 2019; Park et al., 2020). Camera system noise can be categorized into several types, including thermal noise caused by random fluctuations in the electrical charge generated by the image sensor due to heat (Berthelon et al., 2018), and shot noise caused by the random nature of the way light interacts with the image sensor (Roussel et al., 2018). To address the camera system noise, this section simulates camera system noise using probability models with trainable parameters.

Since thermal noise is a form of Gaussian noise, it can be modelled with zero-mean normal distribution $N(0, \sigma)$ where the standard variance σ is considered as the noise level (Denk and Winkler, 2007; Chen, 2021). As for the shot noise, it arises due to the random nature of the arrival times of particles such as photons at the sensor, hence it can be modelled with Poisson distribution (Roussel et al., 2018; Arad et al., 2022) $P(\nu)$ where ν is the noise level. Higher ν values indicate a stronger signal and lower noise, while lower ν values indicate a weaker signal and higher noise. In summary, the noisy RGB can be represented as:

$$rgb_{noisy} = \mu P((N(0, \sigma) + rgb_{clean}) \times \nu)/\nu \tag{6.4}$$

where the brightness factor μ adjusts the intensity of the image with the average scene brightness assumption. The noise level ν, σ are tuned and justified in Section 6.6.2. In practice, the uniform brightness factor is redundant when the ground truth images and recovered images are normalised into the range [0,1]. The comparison between using [0,1] normalisation and using brightness factor is in Section 6.6.

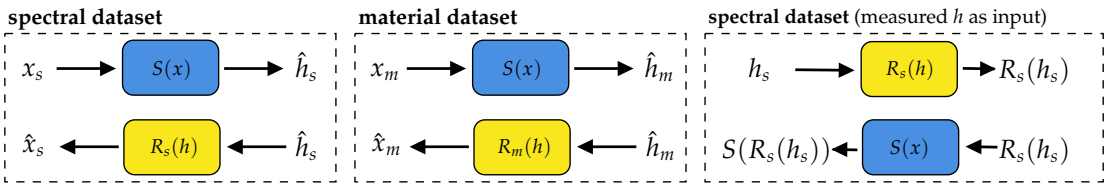


FIGURE 6.5: The data flow for spectral as well as material datasets. Specifically, these two datasets share the same $S(x)$ but have their own $R(h)$ to model the cameras.

6.3.1.3 Other Components and Compression Noise

In a typical in-camera processing pipeline, the final sRGB image is derived through various processing steps from the initial noisy raw image rgb_{noisy}. These steps typically involve operations such as white balance, colour manipulation (Tseng et al., 2022), gamma correction, and JPEG compression (Prakash et al., 2022). While the sRGB gamma correction follows a known equation and JPEG compression can be approximated mathematically with differentiable operations (Wang et al., 2022; Mishra et al., 2022; Shin and Song, 2017), the specific image style configurations for colour manipulation vary across different camera models and are often intricate and proprietary. To avoid the explicit modelling of these camera-specific components applied to the noisy RGB images, this section proposes a network that encompasses the effects of in-camera processing and the noise induced by JPEG compression. This approach is achieved through the integration of two basic Swin transformer layers (Liu et al., 2021b,a) to post-process the noisy image. The basic Swin layer consists of a window-based self-attention and MLP processing to learn from local regions. The window size is set to 8, as the JPEG compression algorithm typically applies the 2D discrete cosine transform on 8×8 blocks. This framework enables the generation of final sRGB images that faithfully capture the effects of these in-camera processes, providing a more robust representation of the recovered sRGB images. Though the network architecture may sound simple, the experiments in Section 6.6.3 show that a simple network is enough to cope with the in-camera processing and compression noise.

6.3.2 Domain-Aware Network Training

The goal of this chapter is to employ hyperspectral images for material segmentation. As there are no hyperspectral images in material datasets, this section proposes to incorporate the spectra dataset ARAD_1K (Arad et al., 2022), which includes pairs of RGB and hyperspectral images, to jointly train the spectral recovery network $S(x)$ and the RGB transformation network $R(h)$ with the material datasets. This chapter denotes the samples in the spectral dataset as (x_s,h_s), and the samples in the material dataset as (x_m). The total loss is defined as $L_{total} = 10 \times L_{band} + 5 \times L_{rgb} + 5 \times L_{spectral} + 0.5 \times L_{domain}$. Section 6.3.1 has introduced the band loss, while the remaining three terms will be introduced in this section. Since the captured RGB images are normalised into the range

[0,1] before passing through the spectral recovery network $S(x)$, the recovered RGB images by $R(h)$ are normalised into the same range as well before calculating the losses.

6.3.2.1 RGB Recovery Loss

Ideally, an RGB image passing through $S(x)$ and $R(h)$ should be able to recover itself, as shown in Equation 6.5.

$$L_{rgb} = L_{trans} + L_{MSE}(x_s, \hat{x}_s) + L_{MSE}(x_s, R_s(h_s)) \tag{6.5}$$

where $\hat{x}_s$, $\hat{x}_m$ are recovered RGB images, $R_s(h_s)$ takes the measured hyperspectral image as input, and L_{MSE} is the mean squared error. As shown in Figure 6.5, x_s and x_m share the same spectra recovery network $S(x)$. As for the RGB transformation networks $R_s(h)$ and $R_m(h)$, these two datasets have their own trainable parameters since the RGB images are taken by different cameras.

6.3.2.2 Spectral Recovery Loss

To calculate the spectral recovery loss term $L_{spectral}$, MatSpectNet compares the recovered hyperspectral image $\hat{h}_s$ with its ground truth h_s, using the mean relative absolute error (L_{MRAE}) (Arad et al., 2022):

$$L_{spectral} = L_{MRAE}(h_s, \hat{h}_s) + L_{MRAE}(h_s, S(R_s(h_s))) \tag{6.6}$$

The two terms correspond to the left and right parts of Figure 6.5. The second term $S(R_s(h_s))$ reverses the order of operations in Figure 6.5 to ensure that the networks $S(x)$ and $R(h)$ are order-irrelevant. Moreover, this term also confirms that the recovered RGB image can be successfully transformed back to the original hyperspectral image.

6.3.2.3 Domain Discrimination Loss

In order to align the features learned from two different datasets, the PatchGAN domain discriminator f_d (Chen et al., 2022) is used. The discriminators $f_{d-spectral}, f_{d-rgb}$ ensure that the hyperspectral images ($h_s, \hat{h}_m$) and the RGB images ($R_s(h), x_m$) are domain-indistinguishable. When training the networks $S(x), R(h)$, the discriminators should not be updated and the loss term L_{domain} should reflect how successfully the network aligns the predictions:

$$L_{domain} = L_{MSE}(f_{d-spectral}(\hat{h}_m), 1) + L_{MSE}(f_{d-rgb}(R_s(h)), 1) \tag{6.7}$$

When training the discriminators, the loss L_{domain} should be configured to identify the sample domain:

$$\begin{aligned} L_{domain} = & L_{MSE}(f_{d-spectral}(h_s), 1) \\ & + L_{MSE}(f_{d-spectral}(\hat{h}_m), 0) \\ & + L_{MSE}(f_{d-rgb}(x_m), 1) \\ & + L_{MSE}(f_{d-rgb}(R_s(h)), 0) \end{aligned} \tag{6.8}$$

This chapter uses the label '1' to indicate the measured sample and '0' to indicate the reconstructed sample.

6.3.3 Interpretable Hyperspectral Processing

The recovery of hyperspectral images opens up the possibility of understanding the contribution of each wavelength to the material segmentation task. This information is useful for modifying camera response curves to generate task-specific images (Saragadam and Sankaranarayanan, 2020). To facilitate this, this section proposes a wavelength-wise self-attention module, named 'spectral attention', which processes the hyperspectral images using predicted filters that have a similar physical meaning as RGB response curves. By aggregating the hyperspectral images based on their dependencies at each wavelength, this module can identify the spectral information that is most relevant to the task at hand.

The spectral attention module computes spectral filters using an attention mechanism applied to the channels of the recovered hyperspectral images. As depicted in Figure 6.6, the hyperspectral image $h \in \mathbb{R}^{n\_bands \times H \times W}$ is reshaped and permuted into $h' \in \mathbb{R}^{HW \times n\_bands}$ to enable the application of self-attention to the spectral channels. Specifically, the spectral attention module employs linear projection to generate the query, key, and value matrices $Q, K, V \in \mathbb{R}^{HW \times n\_bands}$ from the n_bands hyperspectral image h' (Cai et al., 2022c). Then the self-attention output $A \in \mathbb{R}^{HW \times n\_bands}$ is acquired with Equation 6.9:

$$A = VSoftMax(K^T Q) \tag{6.9}$$

where the $SoftMax$ function is applied to the spectral channel. The output A is then scaled to the range [0,1] with the min-max normalisation to align its physical meaning with the RGB response curves, which represent the sensitivity to each wavelength. In practice, this module uses n spectral attention modules in parallel to obtain n filters and constructs an n-channel material image that contains dominant information for the segmentation task. The filters are analysed in the experiment section.

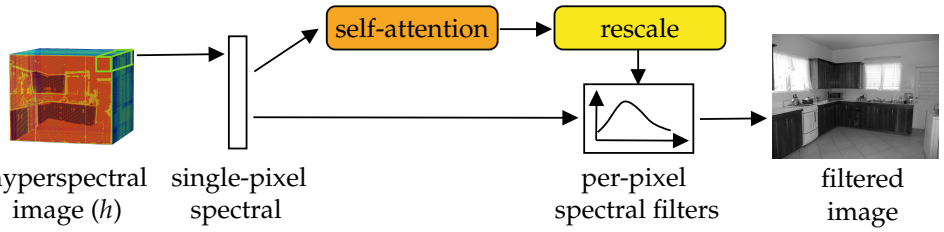


FIGURE 6.6: The network architecture to learn filters to aggregate spectra information.

6.4 Multi-Modal Fusion

In addition to the spectral information learned from the ARAD_1K dataset, this chapter also wants to incorporate human material observations from the spectraldb dataset into the segmentation process. To query the observations, the per-pixel spectral measurements s are used as the bridge to link these two datasets. It is worth noting that the spectral camera and spectrophotometer have different measurement precisions, despite measuring the same physical property. Therefore, in this work, the difference is compared with the spectra shape matrix S of the spectra measurements. The matrix element S_{λ_a,λ_b} is defined in Equation 6.10:

$$S_{\lambda_a,\lambda_b} = |s_{\lambda_a} - s_{\lambda_b}| \tag{6.10}$$

where λ_a and λ_b are the wavelength bands within the range [400nm, 700nm]. This section constructs the spectra shape matrix for both the recovered hyperspectral images as well as the spectraldb measurements (Jakubiec, 2022), which contains multiple spectra measurements indexed by k.

As shown in Figure 6.7, for each pixel i, j in the hyperspectral image, this method finds the matched measurement $S_k^* = \arg\min_{S_k} ||S[i,j] - S_k||_2$ with the L_2 distance. Then the corresponding observations including reflectance, specularity and roughness are appended to pixel i, j of the recovered hyperspectral image. The queried observations together with the filtered features are passed into MLP to extract material features, as shown in Figure 6.1. The way this chapter generates the final material prediction is described in Section 6.5.

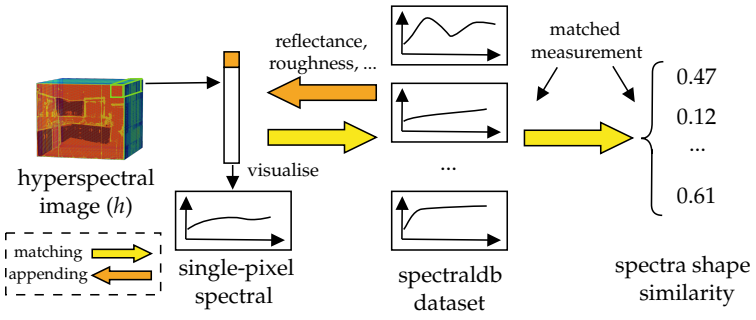


FIGURE 6.7: The mechanism to find the matched sample in the spectraldb dataset.

6.5 Network Training Configuration

The training of MatSpectNet consists of three phases: 1) the pre-training of the spectral recovery network $S(x)$. 2) the training of the physically-constrained recovery network $S(x), R(h)$. 3) the training of the material segmentation decoder. This chapter will illustrate all three components in the following sections.

6.5.1 Data Preparation

This chapter follows the dataset split method as CAM-SegNet and DBAT to prepare both LMD (Schwartz and Nishino, 2020) and OpenSurfaces (Bell et al., 2013a). The models are monitored to choose the best parameters with the validation set and the reported performances are evaluated on the test set.

6.5.2 Pre-training of the Spectral Recovery Network

The MST++ (Cai et al., 2022c) is selected as the spectral recovery network $S(x)$ to process the ARAD_1K RGB images, which are normalised to a range of [0,1] using min-max normalisation. Specifically, for each channel in the RGB image, it subtracts the minimum value and divides the range. The corresponding hyperspectral image remained unchanged. Next, the samples are randomly cropped into 128 $\times$ 128 patches and the images are augmented with random vertical and horizontal flips. To optimise the network, the mean relative absolute error (L_{MRAE}) (Arad et al., 2022) is chosen and the Adam optimiser is used with parameters $\alpha = 4e^{-4}$, $\beta_1 = 0.9$, and $\beta_2 = 0.999$ where α is the initial learning rate. The cosine annealing learning rate scheduler is used with a minimum learning rate of $1e^{-6}$ and the network is trained for 400 epochs with a batch size of 4 per GPU. The best model is selected based on its ability to produce the minimum L_{MRAE} calculated with the validation set.

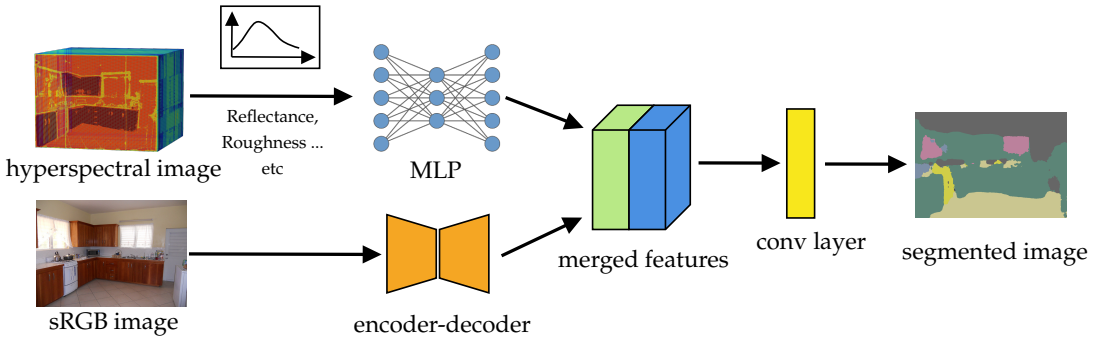


FIGURE 6.8: Feature merging for material segmentation.

6.5.3 Training of the Physically-Constrained Spectral Recovery Network

This section outlines the methodology for training the proposed physically-constrained spectral recovery network. The pre-trained $S(x)$ is loaded and tuned in this section. The AdamW optimiser is adopted with an initial learning rate of $1e^{-6}$, and the values of β_1 and β_2 are set to 0.9 and 0.999, respectively. To avoid making significant alterations to the RGB response curves, the $\Delta band_\lambda$ is cropped into the range [-0.003, 0.003]. Additionally, the linear learning rate scheduler is employed to adjust the learning rate during training. This step is trained for 400 epochs and the parameters that produce the minimum L_{trans} elaborated in Equation 6.1 are kept for material segmentation.

6.5.4 Training of the Material Segmentation Decoder

Figure 6.1 depicts the recovered hyperspectral image $\hat{h}$, which is processed with a MLP for material segmentation. This section provides a detailed figure of the overall architecture by illustrating a material segmentation head, as shown in Figure 6.8. The MLP is configured to generate 128-channel features with hidden units of 64 channels. Recent studies have proposed that for material segmentation tasks, utilising a combination of both material and contextual features, such as features related to objects or scenes, can reduce segmentation uncertainty and lead to improved performance (Heng et al., 2022b,a). As recovered hyperspectral images can provide reliable material features, the MatSpectNet incorporates an additional encoder-decoder segmentation network, the DBAT in Chapter 4, to extract contextual and patch material features from RGB images. The features extracted from both the MLP and the DBAT contain 128 channels each. These features are combined through concatenation and passed into a Conv layer equipped with 3×3 kernels, resulting in the final material label predictions.

During the training of the segmentation head, the parameters of the spectral recovery network $S(x)$ are fixed. To augment the dataset, RGB images are randomly cropped into patches with dimensions of 512×512 and randomly flipped. The AdamW optimizer is utilised with an initial learning rate of $8e^{-5}$, β_1 set to 0.9, and β_2 set to 0.999.

The cyclical learning rate scheduler (Smith, 2017) is applied to gradually decrease the learning rate to $7e^{-7}$ over 400 epochs. The parameters that achieve the highest Pixel Acc are chosen to report the performance for each run.

6.6 Spectral Recovery Experiments

This section analyses the configuration of the proposed RGB transformation network, $R(h)$, by visualising the decay curve of L_{trans} and the recovered sRGB images, to justify the model design and prepare for the material segmentation experiments.

6.6.1 [0,1] Normalisation and Brightness Factor

This section presents the decay curve of L_{trans}, which is introduced in Equation 6.1. Two training configurations are compared, by normalising the RGB images generated by $R(h)$ into the range [0,1] or using a trainable brightness factor. The objective of L_{trans} is to assess the quality of the recovered RGB images $\hat{x}_m$ obtained through the material data flow depicted in Figure 6.5. The normalisation operation acts as an additional constraint in regularising the network $R(h)$. It emphasises that the precise values of individual pixels are not crucial. Instead, the focus is on maintaining the relative values and the range difference of pixel intensities. Since the uniform brightness assumption works by scaling the pixels, it can be redundant when the [0,1] normalisation is used, which also scales the pixels by the range of the pixel values.

The decay curve of L_{trans} provides valuable insights into the impact of the normalisation operation on the training of $R(h)$. As depicted in Figure 6.9, the inclusion of the

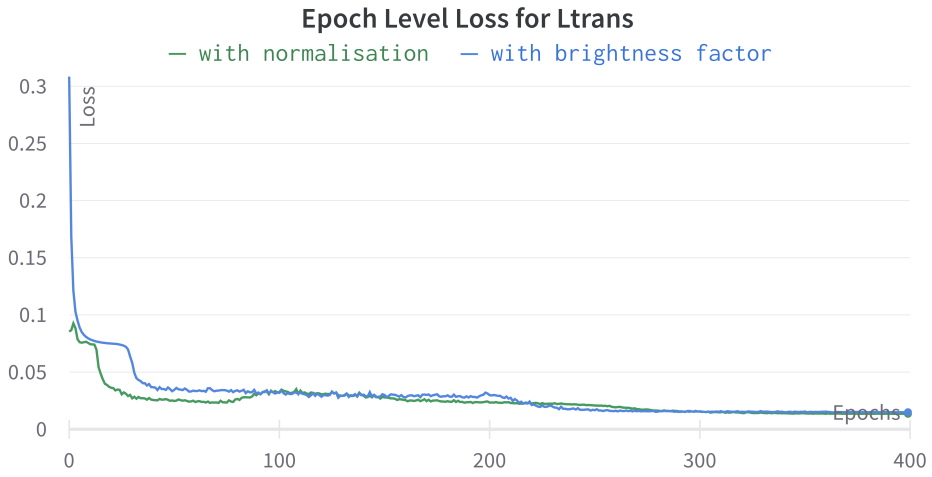


FIGURE 6.9: The loss decay curve of L_{trans} with or without normalisation.

noise level	no σ noise	σ ∈[0,0.001]	σ ∈[0,0.002]	σ ∈[0,0.005]
no ν noise	0.006174	0.005990	0.005947	0.005989
ν ∈[700,900]	0.008225	0.008305	0.008407	0.008775
ν ∈[3000,3500]	0.007437	0.007499	0.007609	0.008028
ν ∈[5000,5500]	0.007338	0.007398	0.007508	0.007928

TABLE 6.1: The converged L_{trans} with different noise level configurations.

normalisation step in the training process has a significant impact on the convergence behaviour of $R(h)$. When the RGB images are normalised, the training of $R(h)$ reaches a convergence level of 0.02301 at epoch 71. In contrast, using brightness factor, it takes approximately 216 epochs to achieve the same level of convergence. By utilising the normalisation constraint, the training of $R(h)$ becomes more efficient. This indicates that scaling absolute pixel values is more difficult than learning the range difference and relative values.

6.6.2 Noise Level Tuning

This section tunes the noise level described in Section 6.3.1.2. According to (Can Karaimer et al., 2019), the noise reduction process happens before the in-camera processing, by means of a high-pass filter. In the simplified camera model shown in Figure 6.2, the noise level is tuned to omit the noise reduction process, by minimising the L_{trans} in Equation 6.1 without the network-modelled in-camera processing component. The evaluation with different noise levels are shown in Table 6.1. In the first column where no thermal noise is modelled, it is evident that the presence of Poisson shot noise significantly degrades the quality of the recovered RGB images, even when only a small amount of Poisson noise is applied (corresponding to high noise level ν). Therefore, for the subsequent experiments, the shot noise is excluded, and the thermal noise level, denoted as σ, is constrained within the range of [0, 0.002]. This range represents the only configuration that surpasses the noise-free scenario, ensuring the integration of noise effects while omitting the noise reduction process.

6.6.3 In-camera Processing Network Justification

The loss term of L_{trans} is shown in Figure 6.10. It converges to 0.005550 at epoch 388, which is 6.68% lower than the one without the in-camera processing network. The visual comparison in Figure 6.11 and Appendix A.2 illustrates the difference between the recovered sRGB images, denoted as $\hat{x}_m$, and their corresponding ground truth images, denoted as x_m. Though the loss L_{trans} converges well, and the images look almost the same, the heatmap suggests that further improvements can be made especially for glass regions, probably caused by the uniform brightness assumption. There are two aspects that can be invested to improve the image quality.

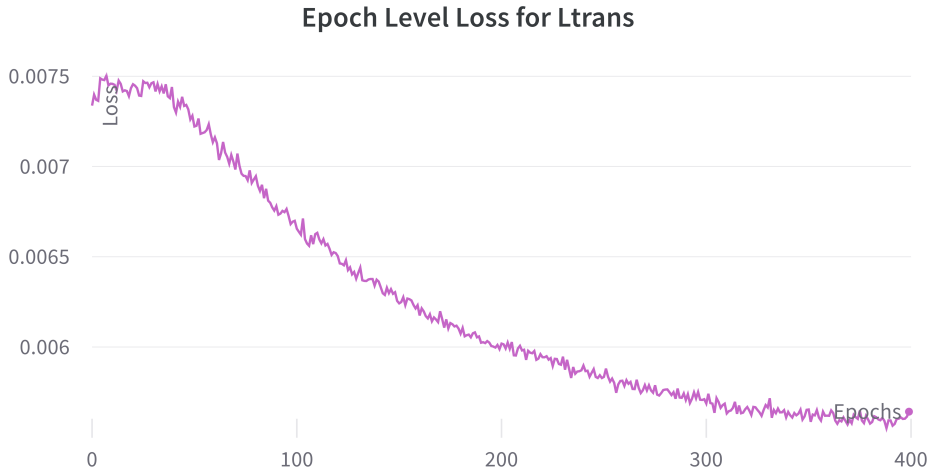


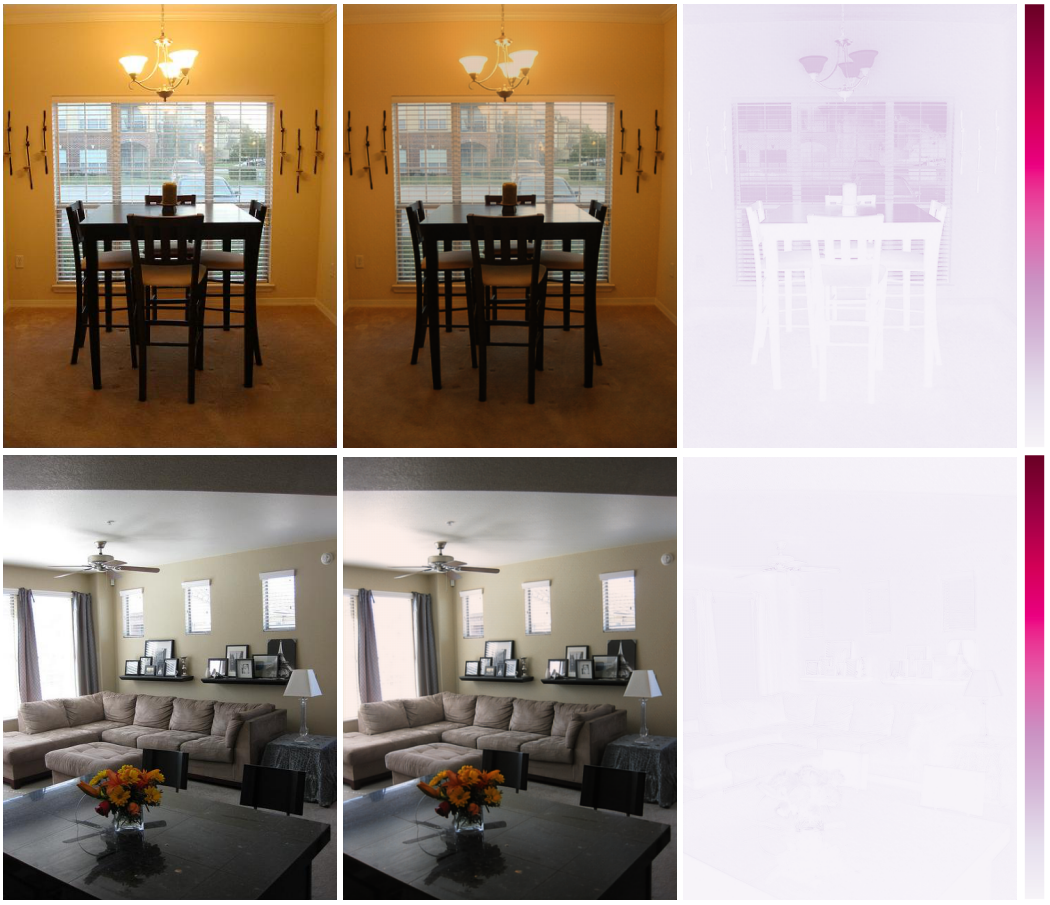
FIGURE 6.10: The loss decay curve of L_{trans} with tuned noise levels and in-camera processing.

Firstly, the field of spectral recovery remains an active area of research. Even with the use of a supervised dataset, the current performance of the spectral recovery network in generating hyperspectral images is not optimal. Through the implementation of an improved spectral recovery network, it is possible to achieve superior results by recovering more accurate and detailed information in the hyperspectral domain.

Secondly, the camera model incorporated in the network $R(h)$ may suffer from the uniform brightness assumption, which does not hold in situations where multiple light sources exist. To address this issue, it is recommended to conduct experiments using more complex brightness estimation methods, such as (Afifi and Brown, 2019). Moreover, it is possible to build a dataset-independent camera model with raw-RGB images instead of sRGB images. This adjustment allows for a simple but realistic camera model that better captures the transformation between hyperspectral and RGB images. Moreover, by eliminating the need for an additional network component to simulate the camera model, it is possible to streamline the image recovery process and potentially achieve improved results.

6.6.4 Spectral Recovery Performance

Since there are no captured hyperspectral images in material datasets, the spectral recovery network is quantitatively evaluated on ARAD_1K. As shown in table 6.2, with the material dataset, the performance of the spectral recovery network $S(x)$ decays. This is expected since the training objective is changed from recovering hyperspectral images in ARAD_1K to material datasets. Luckily, the decay curve of L_{trans} in Figure 6.10 indicates that the quality of recovered hyperspectral images in material datasets is improved. Moreover, in Figure 6.12, with MatSpectNet, the differences between x_m



(A) Ground Truth RGB x_m (B) Recovered RGB $\hat{x}_m$ (C) Heatmap of $|x_m - \hat{x}_m|$

FIGURE 6.11: The visualisation of two pairs of sRGB and recovered sRGB. The heatmap is measured by averaging the difference between normalised (range [0,1]) x_m and $\hat{x}_m$ across R, G, B channels. Pink means the difference is 0, and red means the difference is 1.

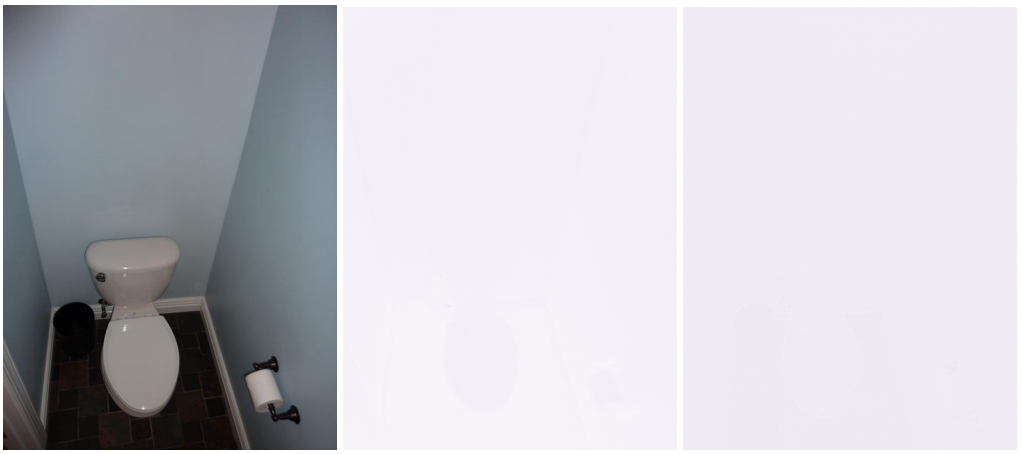
model	MRAE	RMSE
MST++	0.1645	0.0248
MatSpectNet	0.1774	0.0316

TABLE 6.2: The performance of the spectral recovery network $S(x)$, evaluated with metrics in (Arad et al., 2022).

and $\hat{x}_m$ are smaller, supported by the whiter heatmap colour, compared with directly using the pre-trained MST++ (Cai et al., 2022c).

6.6.5 Gradient Magnitude of Recovered Hyperspectral Images

The incorporation of hyperspectral images offers a unique advantage in material segmentation tasks as they capture the portion of light reflected by materials in the scene.



(A) Ground Truth RGB x_m

(B) Heatmap of $|x_m - \hat{x}_m|$ with MatSpectNet

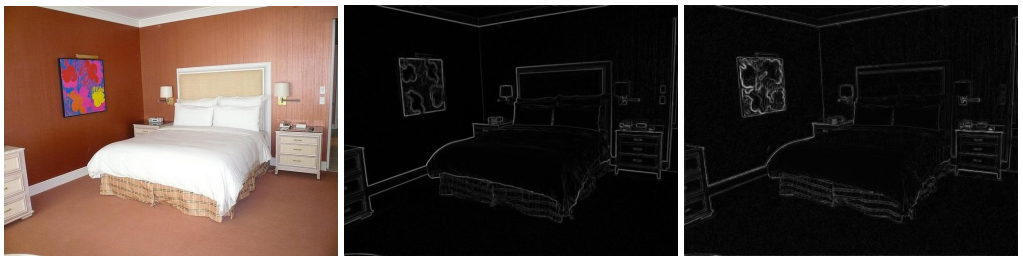
(C) Heatmap of $|x_m - \hat{x}_m|$, with MST++(Cai et al., 2022c)

FIGURE 6.12: Heatmaps of recovered RGB images with MatSpectNet or pre-trained MST++.

As reflectance is a fundamental characteristic of materials, hyperspectral images provide vital discriminative features in addition to conventional RGB images. To validate the efficacy of hyperspectral images in material segmentation, one approach is to examine whether pixels representing the same material exhibit similar spectral profiles within a scene. The gradient magnitude plots of the hyperspectral images at wavelengths 360nm and 700nm, as depicted in Figure 6.13, provide valuable insights in this regard.

For instance, in the bedroom image, at wavelength 360nm, the white quilt, pillow, and yellow headboard are all composed of fabric materials. The gradient magnitude plot shows no distinct boundaries, illustrating that hyperspectral images accurately capture reliable material properties. However, when analysing the painting on the wall at wavelength 700nm, clear boundaries between the colour spots emerge, potentially attributed to the diverse ingredients present in the painting. Similarly, the bed sheet reveals distinct regions, where the material of the strips may differ from the surrounding yellow fabric area. These observations underscore the challenges in annotating material segments and the complexities involved in defining appropriate material categories.

The analysis of hyperspectral image gradients provides compelling evidence for the potential of leveraging hyperspectral images in material segmentation tasks. The capacity to capture distinct spectral profiles of materials aids in accurately delineating material boundaries, thus offering a valuable contribution to the advancement of material segmentation methodologies. Furthermore, it highlights the need for robust annotation techniques and the exploration of innovative strategies for defining meaningful material categories in this challenging domain.



(A) Bedroom Image (B) 360nm (C) 700nm

FIGURE 6.13: The gradient magnitude of the recovered hyperspectral images.

6.7 Material Segmentation Experiments

This section assesses the performance of the network on two material datasets, namely LMD (Schwartz, 2018; Schwartz and Nishino, 2020) and OpenSurfaces (Bell et al., 2013a), using the evaluation metrics reported in (Heng et al., 2022a). The proposed MatSpectNet is trained for 400 epochs with 8 NVIDIA GeForce RTX 3090 GPUs.

6.7.1 Quantitative Evaluation

Table 6.3 presents numerical evaluations of seven networks. The proposed MatSpectNet outperforms all other compared networks on both datasets. Specifically, on LMD, the MatSpectNet achieves the highest Pixel Acc of 88.24% and Mean Acc of 83.82%. Compared to the second-best performing network, DBAT (Heng et al., 2022a), the MatSpectNet shows improvements of 1.60% and 3.42% for Pixel Acc and Mean Acc, respectively. On OpenSurfaces, the MatSpectNet also achieves the best performance with accuracy scores of 87.13% and 71.29% for Pixel Acc and Mean Acc, respectively. The corresponding improvements compared to the second-best scores are 0.99% and 0.86%. In particular, for LMD, the higher improvement in Mean Acc and lower improvement in Pixel Acc indicate that the MatSpectNet improves the performance for hard-to-recognize material categories compared to other networks, suggesting that hyperspectral information can provide reliable material features. The detailed per-category performance is reported below.

Datasets	LMD		OpenSurfaces			–		
Architecture	Pixel Acc	Mean Acc	Pixel Acc	Mean Acc	mIoU	#params (M)	#flops (G)	FPS
ResNeSt-101 (Zhang et al., 2022)	82.45 ± 0.20	75.31 ± 0.29	85.10	67.13	55.32	48.84	63.39	25.57
EfficientNet-b5 (Tan and Le, 2019)	83.17 ± 0.06	76.91 ± 0.06	84.63	65.47	53.25	30.17	20.5	27.00
Swin-t (Liu et al., 2021b)	84.70 ± 0.26	79.06 ± 0.46	86.19	69.41	57.71	29.52	34.25	33.94
CAM-SegNet-DBA	86.12 ± 0.15	79.85 ± 0.28	86.64	69.92	58.18	68.58	60.83	17.79
DBAT	86.85 ± 0.08	81.05 ± 0.28	86.28	70.68	58.08	56.03	41.23	27.44
MatSpectNet	**88.24 ± 0.10**	**83.82 ± 0.23**	**87.13**	**71.29**	**58.92**	57.7	42.16	26.83

TABLE 6.3: The performance (in percentage) inherited from Chapter 4 reported on the LMD and the OpenSurfaces. The FPS value of MatSpectNet is calculated by processing 1000 images with one NVIDIA 3090. The uncertainty evaluation is reported across five independent runs.

Model	MatSpectNet	DBAT	ResNeSt-101	EfficientNet-B5	Swin-t	CAM-SegNet-DBA
Asphalt (4.87)	91.57 ± 0.23	88.66 ± 0.72	**94.35 ± 0.27**	82.17 ± 2.80	91.83 ± 1.09	89.87 ± 1.94
Ceramic (2.95)	**77.81 ± 1.07**	68.31 ± 1.31	62.86 ± 0.67	73.34 ± 0.42	75.35 ± 0.42	75.01 ± 0.64
Concrete (4.73)	65.79 ± 1.30	66.90 ± 1.07	60.53 ± 2.00	59.36 ± 2.98	57.42 ± 4.88	**69.20 ± 2.81**
Fabric (10.96)	**93.23 ± 0.27**	93.14 ± 0.16	86.420 ± 0.92	85.33 ± 0.20	88.71 ± 0.50	90.79 ± 0.43
Foliage (15.43)	**96.24 ± 0.15**	95.35 ±0.12	91.25 ± 1.16	88.21 ± 0.32	95.57 ± 0.45	94.04 ± 0.79
Food (10.03)	95.37 ± 0.18	93.27 ± 0.22	94.96 ± 0.34	**95.84 ± 0.14**	92.51 ± 0.83	95.19 ± 0.24
Glass (1.94)	73.03 ± 0.83	73.27 ± 0.67	68.33 ± 0.34	77.83 ± 0.94	77.95 ± 0.99	**84.88** ± 1.11
Metal (6.17)	80.67 ± 0.45	79.99 ± 0.51	80.66 ± 0.34	76.67 ± 0.28	81.54 ± 1.36	**81.83 ± 0.48**
Paper (1.76)	**78.35 ± 1.37**	73.83 ± 0.67	71.14 ± 1.99	77.21 ± 0.13	63.05 ± 1.90	66.48 ± 1.43
Plaster (2.54)	**82.36 ± 1.95**	71.43 ± 0.71	78.76 ± 0.62	73.11 ± 0.64	78.12 ± 1.90	72.37 ± 1.03
Plastic (1.47)	**65.61 ± 2.51**	50.62 ± 1.45	36.07 ± 3.42	39.59 ± 0.64	51.64 ± 1.31	52.07 ± 2.28
Rubber (1.08)	81.57 ± 1.42	82.61 ± 1.01	79.57 ± 1.62	69.73 ± 0.29	**83.48 ± 0.67**	81.63 ± 1.79
Soil (8.22)	**85.53 ± 0.47**	84.25 ± 0.50	73.15 ± 2.67	79.73 ± 0.55	76.89 ± 1.11	80.39 ± 1.73
Stone (3.13)	82.06 ±1.28	**86.94 ± 0.95**	52.12 ± 0.93	70.07 ± 0.76	73.05 ± 1.92	60.73 ± 2.76
Water (11.17)	96.74 ± 0.19	**97.12 ± 0.10**	97.54 ± 0.28	95.30 ± 0.32	95.78 ± 0.70	94.95 ± 0.69
Wood (13.54)	**94.47 ± 1.08**	90.53 ± 0.37	76.71 ± 1.23	86.69 ± 0.24	82.03 ± 1.11	87.63 ± 0.98
Pixel Acc	**88.24 ± 0.10**	86.85 ± 0.08	82.45 ± 0.20	83.17 ± 0.06	84.71 ± 0.26	86.12 ± 0.15
Mean Acc	**83.82 ± 0.23**	81.05 ± 0.28	75.31 ± 0.29	76.91 ± 0.06	79.06 ± 0.46	79.85 ± 0.28

TABLE 6.4: Per-category performance analysis Heng et al. (2022a). The best performance achieved for each category is written in bold text, and the numbers are reported in percentage. The number after the material category is the pixel coverage (in percentage) of each material in the dataset.

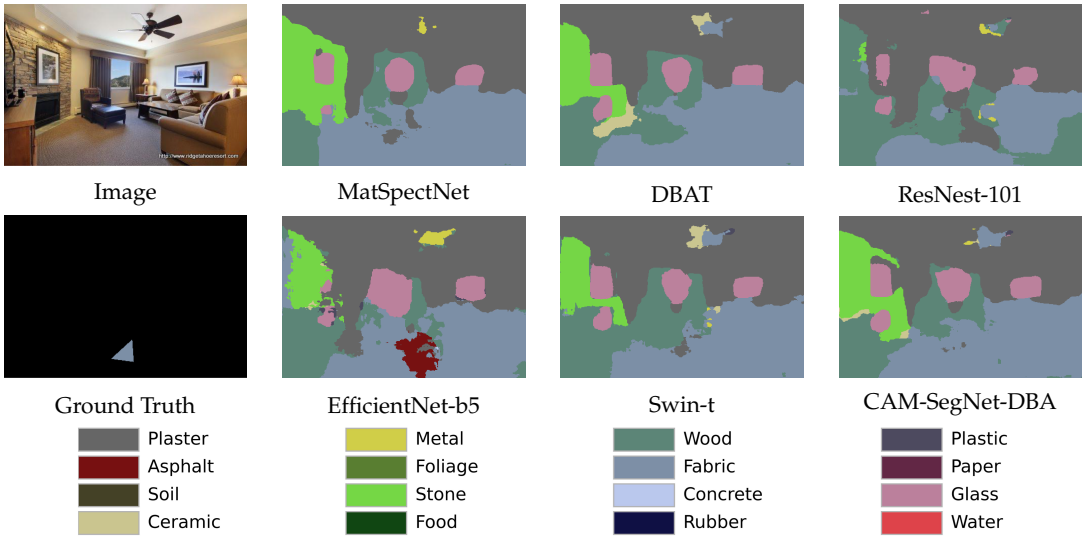


FIGURE 6.14: Predicted segmentation of one living room image.

Table 6.4 displays the per-category performance of the proposed MatSpectNet and other networks. Notably, the model outperforms other models in eight categories, particularly those covering a small portion of the annotated samples, such as paper and plastic, as indicated by the numbers next to the category names. This suggests that the recovered hyperspectral images provide informative material features that enhance performance, even when the number of annotations is limited.

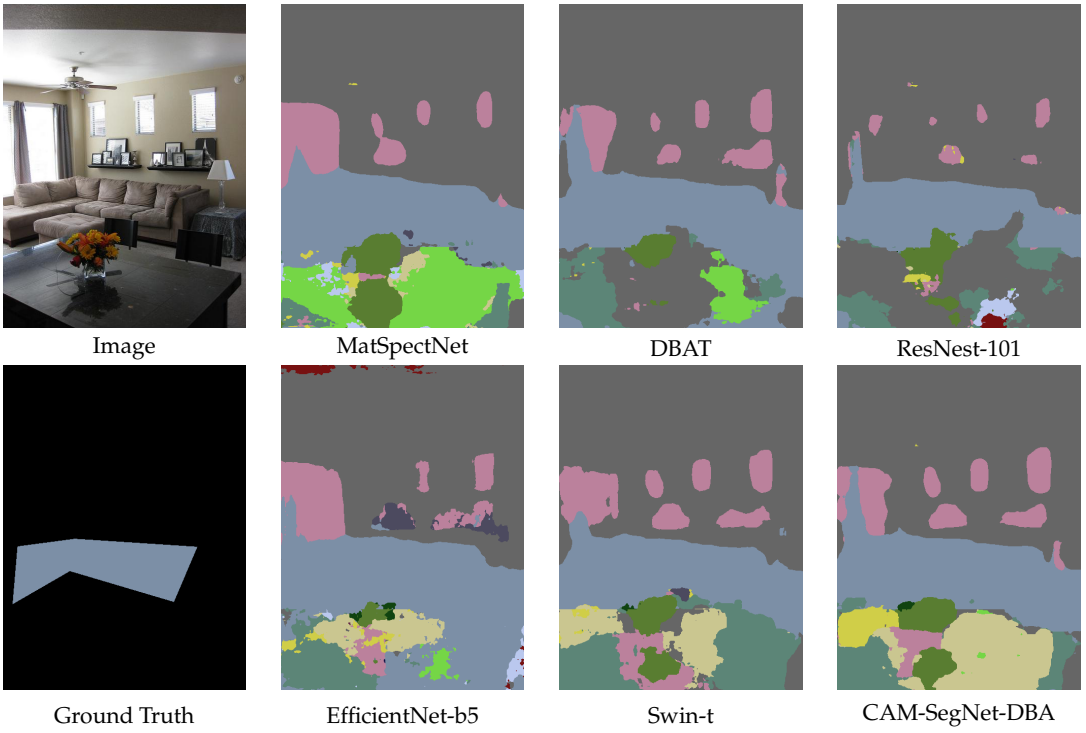


FIGURE 6.15: Predicted segmentation of another living room image.

6.7.2 Qualitative Evaluation

The segmented images are displayed in Figure 6.14. Despite the sparsely labelled ground truth (Heng et al., 2022b,a; Schwartz and Nishino, 2020), the MatSpectNet is the only model that can accurately classify nearly all the pixels belonging to the stone wall, even those under bright lighting conditions. This suggests that the calibrated spectral information can boost network performance under varying illumination conditions, as compared to other purely visual-based networks.

Another segmentation visualisation is presented in Figure 6.15. The table shown in the image lacks distinguishable texture, making it difficult even for humans to determine its material. As a result, in all other five models, the segmented images appear messy near the table region. The DBAT model struggles to distinguish between plaster and wood, whereas models such as Swin-t (Liu et al., 2021b) and EfficientNet (Tan and Le, 2019) are affected by reflected light and incorrectly recognize parts of the table as ceramic or glass. This indicates that RGB image does not provide reliable visual features for material segmentation task. In contrast, the MatSpectNet model avoids such noisy recognition and confidently identifies the table as stone. This indicates that the recovered hyperspectral image is a robust description of materials, irrespective of the illumination.

However, it is also noticed that the material labels defined in LMD lack the ability to fully describe indoor scenes in a dense manner. Furthermore, the similarity in the

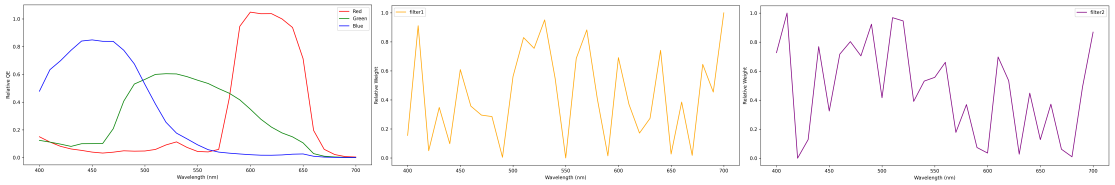


FIGURE 6.16: The filter weight of two filters for each wavelength.

appearance of different materials makes the segmentation task challenging to solve with RGB images. As spectral information is independent of visual appearance, it is possible that hyperspectral reconstruction could be a potential solution to overcome the limitation of sparse labelling for pixels that cannot be adequately described by the given labels.

6.7.3 Filter Analysis

Section 6.3.3 introduces filters that aggregate spectral information using a weighted sum. These filters are designed to behave similarly to RGB filters, where the filter values represent the importance of each wavelength. In order to gain a better understanding of how these filters work, Figure 6.16 plots two of the filter weights for each wavelength alongside the RGB response functions. The filters show wavelength ranges where the weights are larger than others, indicating the wavelengths that the network depends on, such as the range between 420 and 560 nm for filter 2. Moreover, unlike the RGB response functions, the learned filters can have more than one important region, as demonstrated by filter 1. This finding suggests that aggregating both short and long-wavelength information in the same filter is beneficial for material segmentation. Further investigation into the optimal weighting of different wavelength ranges can be a potential avenue for future research. Lastly, it is worth noting that the learned filters in the network exhibit a tendency to assign weights close to 0 for frequencies that are near those with large weights, such as 490nm and 550nm. This behaviour suggests that the training process eliminates frequencies that do not contribute significantly to material segmentation. One possible explanation for this phenomenon is that the spectral profiles often exhibit flat regions, as depicted in Figure 6.17. The filters effectively encode the spectral curve by sampling a few points from these flat regions and primarily focusing on the regions where the intensity varies.

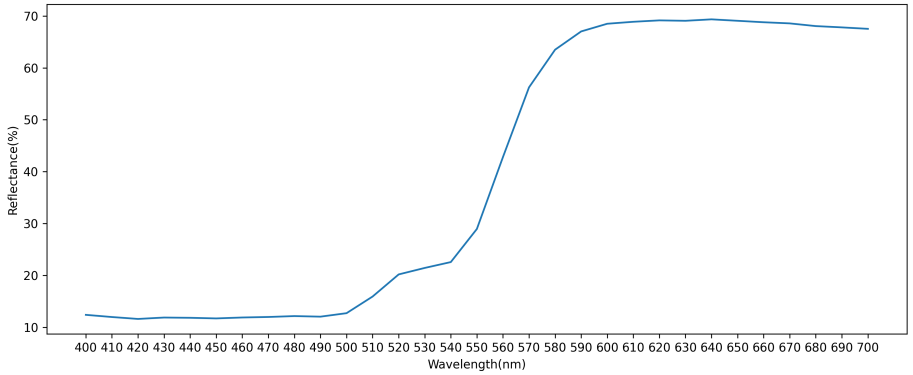


FIGURE 6.17: A sample of the recovered spectral profile for a pixel classified as plastic.

6.8 Ablation Study

This section assesses the impact of each component of the proposed MatSpectNet by introducing three additional models of increasing complexity, as shown in Table 6.5. For each model, the number of filters is adjusted in the interpretable hyperspectral processing module and report the corresponding (Pixel Acc/Mean Acc) results on LMD in Table 6.6.

6.8.1 Regularised Spectral Recovery

Network 1 utilises a pre-trained spectral recovery network to generate the hyperspectral images. However, since the network is not specifically tuned for the material datasets, the network trained on the ARAD_1K dataset is not generalisable, and the

Methods	Regularised Spectral Recovery	Domain Alignment	Multi-Modal Fusion
Network 1			
Network 2	✓		
Network 3	✓	✓	
MatSpectNet	✓	✓	✓

TABLE 6.5: The network variations and the corresponding enabled components.

n_filters	4	8	12	16
Network 1	86.73/80.91	86.55/80.14	86.94/81.13	87.08/81.20
Network 2	86.93/81.41	87.17/81.74	87.69/81.92	87.10/81.63
Network 3	87.16/82.88	87.22/82.93	87.46/81.71	87.54/82.36
MatSpectNet	87.95/83.17	87.66/83.14	**88.24/83.82**	88.07/83.52

TABLE 6.6: The network performance for each variation against four filter number choices. The numbers corresponds to Pixel Acc/Mean Acc in percentage.

$S(x)$	Pixel Acc	Mean Acc
Fine-tune	81.91 ± 0.22	75.35 ± 0.31
Frozen	88.24 ± 0.10	83.82 ± 0.23

TABLE 6.7: The network performance evaluated by fine-tuning or freezing the parameters of $S(x)$. The evaluations are reported by training the models for five times.

network performance is hardly improved. In contrast, when the spectral recovery network is equipped with the physically-constrained RGB transformation network, Network 2 achieves up to 0.75 p.p. improvement in Pixel Acc and 1.60 p.p. improvement in Mean Acc. However, the training process does not guarantee an increase in network performance, as evidenced by the results for filter number 8.

6.8.2 Domain Alignment

The domain alignment training strategy is designed to learn generalisable features for material datasets without any training samples. The reported evaluations show that network 3 improves network performance for all filter choices, indicating that the features extracted by the spectral recovery network are well-suited for material datasets.

6.8.3 Multi-Modal Fusion

The spectradb dataset provides measurements and observations that can help reduce network uncertainty in identifying materials. As shown in Table 6.6, the MatSpectNet achieves the best performance with an improvement of 0.78 p.p./2.11 p.p. when the filter number is set to 12 compared with Network 3. The significant increase indicates that human observations, such as roughness, can reliably differentiate between different materials.

6.8.4 Tune the Spectral Reconstruction Parameters

The experiments reported until now all freeze the parameters of the spectral recovery network when training for the material segmentation. This ensures that the recovered hyperspectral images are used for material segmentation. However, it is conceivable that a more favourable performance could be achieved through unconstrained fine-tuning. Therefore, the evaluations by fine-tuning or freezing the parameters of $S(x)$ are reported in Table 6.7. By fine-tuning the parameters, the network performance drops by 6.33 p.p. in Pixel Acc, and drops by 8.47 p.p. in Mean Acc. This indicates that tuning of $S(x)$ together with the material segmentation breaks the spectral recovery network.

6.9 Conclusion

This section introduces the MatSpectNet, a model which employs a physically-constrained spectral recovery network to segment materials using reconstructed hyperspectral images. To leverage the existing spectral recovery dataset, domain-aware discriminators are used to align the material dataset and enhance the quality of the reconstructed hyperspectral image. This chapter also incorporates material observations, such as roughness, to improve the reliability of material predictions. The experiments demonstrate that the proposed MatSpectNet outperforms existing models and can handle images captured under varying lighting conditions. As a pioneering attempt towards explicit material feature extraction, the MatSpectNet is still imperfect. For example, as shown in Figure 6.8, the inputs of the spectral recovery network and the DBAT are the same RGB image. Since the recovered hyperspectral image should contain richer information compared with RGB image, it would be possible to merge the RGB feature extraction branch into the hyperspectral recovery branch. As my PhD study approaches its conclusion, potential improvements such as this are left as future work.

Chapter 7

Conclusion and Future Work

This chapter marks the conclusion of my PhD study (Section 7.1 to Section 7.4) and offers potential future research directions for scholars. Throughout this research project, three and half study contributions have been presented exploring two main research topics with the goal of achieving high-performance and real-time material segmentation (24FPS, 80% accuracy). The first topic focuses on the implicit extraction of material features from image patches, while the second topic delves into the explicit extraction of material features from recovered hyperspectral images. In Section 7.6, the feasible future work is introduced based on my research experience.

7.1 Contribution A: CAM-SegNet

The proposed CAM-SegNet in Chapter 3 tackles the research problem that combining contextual and material features during training requires labels related to objects or scenes. Existing methods (Schwartz and Nishino, 2020, 2016, 2019; Schwartz, 2018) tend to use pre-trained networks targeting object recognition or scene classification to provide contextual cues. However, the pre-trained networks are not fine-tuned with the material segmentation branch, and annotating images requires extra undesired costs. In order to make predictions with both material and contextual features extracted from sparsely labelled material segmentation datasets, the proposed hybrid CAM-SegNet leverages contextual features of material boundaries and a self-training mechanism to achieve accurate segmentation without extra labels or pre-trained networks. The experimental results demonstrate that CAM-SegNet surpasses recently proposed network architectures and single-branch approaches in the control group, achieving a significant improvement of 3-20% in Pixel Acc and 6-28% in mIoU.

7.2 Contribution B.1: DBAT

While CAM-SegNet achieves improved performance, it still suffers from high computing overhead due to the multi-branch architecture. Moreover, the patch resolution is fixed for all images, which may limit the network ability to learn material features. In consideration of these problems, Chapter 4 introduced the DBAT, which enables learning material features from cross-resolution image patches in a single feed-forward propagation. In the initial stage, the images are passed through a transformer backbone that operates on 4×4 patches and progressively enlarges the patch resolution by merging neighbouring patches. Following that, a DBA module is introduced to dynamically combine the intermediate features obtained from cross-resolution image patches. Evaluating the DBAT on the LMD dataset, it achieves a Pixel Acc of 86.85% while maintaining a high frame rate of 27.44 FPS. This performance demonstrates 21.21% improvement over the CAM-SegNet.

7.3 Contribution B.2: Network Interpretability

DBAT has sufficed in the goal of designing a network to achieve high performance in real-time. However, similar to other network-based methods, the DBAT faces challenges in terms of interpretability. Therefore, Chapter 5 delves deeper into interpreting the behaviour of the DBAT using statistical and visual analysis tools. These tools include calculating the attention head equivalent patch resolution, visualising attention masks, and assessing the CKA heatmap, as proposed by Nguyen et al. (2020); Raghu et al. (2021). Additionally, to achieve a more human-readable interpretation of the features, Chapter 5 also employs the network dissection method introduced by Zhou et al. (2018); Bau et al. (2017, 2019, 2020), which aligns layer neurons with semantic concepts.

Through the analysis of the semantic concepts associated with the extracted features, the experiments illustrate the proficiency of DBAT in capturing material-related features, such as texture, which is crucial for distinguishing between different materials. Furthermore, by comparing the semantic concepts of features extracted by other networks trained with either material or object datasets, the results also highlight the influence of network architecture on the extracted features. This analysis confirms the effectiveness of the patch-based design in driving the networks to perform segmentation based on material features.

7.4 Contribution C: MatSpecNet

The CAM-SegNet and DBAT are designed to learn material-related features from RGB image patches. With further research, it becomes evident that this approach is implicitly based and does not provide a definitive assurance that the model can effectively learn local features, as indicated in Section 5.4. Therefore, Chapter 6 proposed a network named MatSpectNet to explicitly learn material features from material optical properties measured by hyperspectral cameras (Behmann et al., 2018). In the initial stage, the network is trained to reconstruct hyperspectral images by leveraging the physical model of the camera as a constraint, ensuring the preservation of the intrinsic physical properties of the hyperspectral images. Subsequently, the recovered hyperspectral images are processed through a MLP to extract material features based on the spectral information associated with each individual pixel. This approach allows for the explicit learning of material features from spectral data, enhancing the ability of networks to discern and represent different materials accurately.

The proposed network, MatSpectNet, outperforms existing models in the material segmentation task. With a Pixel Acc of 88.24% and a Mean Acc of 83.82%, the network shows an improvement of 1.60%/3.42% over the DBAT. Notably, MatSpectNet is particularly adept at recognising material categories that have limited samples, even under varying light conditions.

7.5 Limitations of the Research

While these contributions represent advancements in the field of computer vision and material segmentation, there is still much work to be done. Listed below are the following limitations of the studies described in this research project:

1. The sparsely labelled open-access material segmentation datasets poses challenges in ensuring the reliability of numerical evaluations. The advancements made in unlabelled areas may not be adequately reflected in the numerical evaluation metrics. For instance, Figure 6.15 showcases a well-segmented table using MatSpectNet; however, the absence of ground truth annotations prevents readers from accurately quantifying the true improvements in terms of metrics like Pixel Acc or Mean Acc. As a result, the network performance might be underestimated, emphasising the need for alternative evaluation approaches to capture the advancements achieved in unlabelled regions.

2. The network dissection method, while valuable, possesses limitations in terms of the number of semantic concepts it can effectively analyse. Expanding the coverage to incorporate additional concepts necessitates the availability of further

labelled samples. Among the limited set of five concepts, only texture and colour are directly relevant to material characteristics. This constraint undermines the comprehensiveness of network interpretability since these five concepts are insufficient to encompass the vast array of features the network is capable of learning. Furthermore, it should be noted that interpretation is only feasible for disentangled neurons. This challenge underscores the pressing need for more robust and comprehensive network interpretability methods to gain a deeper understanding of the learned features.

3. The scope of this research is centred around indoor scenes, and the evaluation is confined to two specific datasets, namely LMD and OpenSurfaces. It is worth noting that densely labelled datasets already exist for material segmentation in outdoor or geoscience scenarios. To broaden the applicability and robustness of material segmentation networks, future endeavours can involve redesigning the networks to encompass the common characteristics of materials across various scenes. The development of more robust and versatile segmentation models can be pursued, which would allow for evaluation on more reliable and diverse datasets. Such an approach would contribute to the advancement of material segmentation in a wider range of scenarios and enhance the generalisability of the proposed models.

4. This study primarily focuses on the generation of material labels. In certain applications, such as immersive sound rendering, material labels are not only used for visual purposes but also play a crucial role in connecting with acoustic properties within the rendering engine. However, it is worth noting that the categories produced by the network may not align precisely with the labels supported by the rendering engine. This misalignment has the potential to detrimentally affect the performance of downstream tasks that rely on accurate correspondence between material labels and acoustic properties. To mitigate this issue, it is highly recommended to construct dedicated datasets specifically tailored for tasks that depend on material labels. One approach could involve annotating RGB images with acoustic properties measured using specialised equipment, rather than relying solely on human-labelled material categories. By incorporating such dedicated datasets, which provide more precise and reliable information about the acoustic properties associated with material labels, the performance and compatibility of downstream tasks can be significantly improved.

5. Regarding the MatSpectNet, it is a hybrid network that incorporates an additional branch for hyperspectral image recovery, while the other branch continues to learn from the original RGB images. The recovered hyperspectral images are expected to contain more comprehensive information compared to RGB images since hyperspectral images can be transformed into RGB images using known response functions and in-camera processing. This leads to the possibility of

combining the two branches into a unified approach. Furthermore, the camera model employed in this network relies on network blocks to implicitly simulate certain modules within the in-camera processing pipeline. It is important to find the appropriate balance for these network blocks. If the network block is excessively powerful, the explicitly modelled components may lose their significance. Conversely, if the network block is overly simplistic, it might fail to implicitly model the missing components. Hence, it is crucial to propose a simpler and more straightforward method to leverage hyperspectral images effectively. For instance, one potential approach could involve directly labelling material categories on existing hyperspectral images, and ignore the recovery branch. By developing such a method, the benefits of hyperspectral images can be maximised while circumventing potential issues associated with the network complexity. This would pave the way for improved material segmentation performance and facilitate a more straightforward integration of hyperspectral information into the segmentation process.

7.6 Future Works for Material Segmentation

In this section, three promising avenues for future research are proposed, aimed at providing potential successors with valuable directions. These include:

1. **Refinement of Network Architecture**: Building upon the foundation laid by MatSpectNet, there is an opportunity to explore and develop a revised network architecture. This could involve optimising the design and configuration of the network to enhance its performance and efficiency in material segmentation tasks.

2. **Data Synthesis Strategy**: Developing a data synthesis strategy holds great potential for generating paired RGB and hyperspectral image datasets. This approach could involve leveraging advanced techniques, such as simulation and PBR, to create synthetic datasets that closely resemble real-world scenarios. This synthesised data would enable comprehensive training and evaluation of material segmentation models.

3. **Annotating Sparsely Labelled Datasets**: To address the limitations imposed by sparsely labeled datasets, it is important to devise a pipeline equipped with automatic tools for annotating these datasets more comprehensively. This would involve leveraging SOTA techniques, such as prompt-based giant vision models, to automatically annotate the existing datasets with accurate material regions. By doing so, the utility and effectiveness of datasets for training material segmentation models would be significantly enhanced.

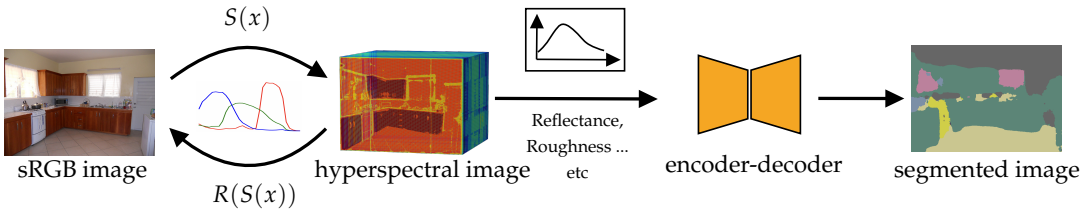


FIGURE 7.1: A unified network architecture that merges the spectral recovery task with the material segmentation network.

Exploring these three potential avenues would contribute to advancing the field of material segmentation, enabling the development of more sophisticated network architectures, improving dataset availability through data synthesis, and enhancing the quality of existing datasets through automated annotation pipelines.

7.6.1 Refinement of Network Architecture

In the pursuit of harnessing the potential of hyperspectral images for material segmentation, the MatSpectNet has demonstrated its effectiveness through comprehensive quantitative and qualitative evaluations presented in Chapter 6. While the MatSpectNet has provided an encouraging step towards the utilisation of hyperspectral images, there remains room for further improvement in its network design.

Notably, existing literature highlights the under-determined nature of the spectral recovery task (Arad et al., 2022; Cai et al., 2022c), characterized by many-to-one mappings between hyperspectral images and RGB images, often referred to as color metamerism (de Abreu Santos PEREIRA et al., 2019). This mapping process inherently involves information loss, as the transformation from hyperspectral images to RGB images inevitably results in the omission of certain details. Consequently, hyperspectral images inherently contain richer information pertaining to the captured scenes in comparison to RGB images.

From the perspective of network architecture, it is thus conceivable to merge the spectral recovery branch with the feature learning branch and facilitate concurrent training of the entire network. This architectural modification presents a plausible approach to integrate spectral recovery and feature learning in a synergistic manner. By doing so, the network can jointly exploit the information-rich hyperspectral images while effectively leveraging the feature learning capabilities. A potential network architecture that encapsulates these improvements is depicted in Figure 7.1. It seamlessly integrates the segmentation network with the spectral recovery network, eliminating the presence of parallel branches.

By incorporating such enhancements into the network design, it is anticipated that the overall performance of material segmentation can be further advanced, capitalizing on

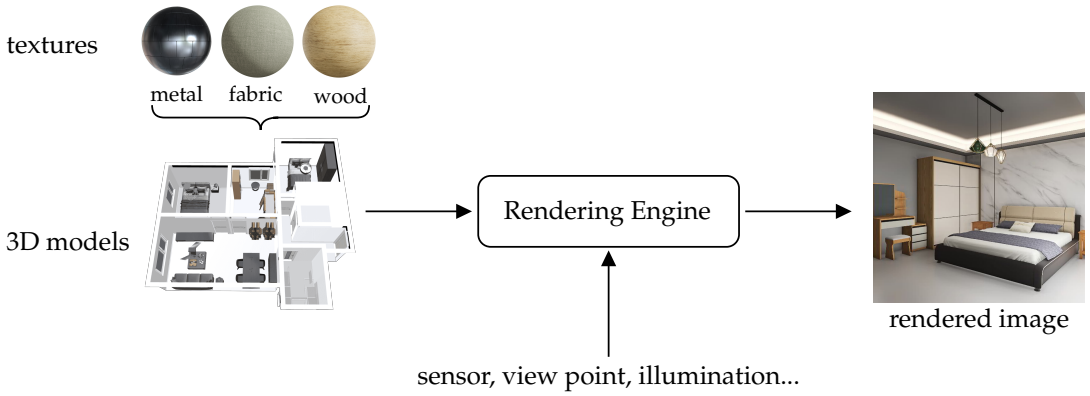


FIGURE 7.2: The pipeline to render the images with 3D models and material textures

the inherent advantages of hyperspectral images and enabling simultaneous spectral recovery and feature learning within a unified framework.

7.6.2 Data Synthesis Strategy

The collection of RGB and hyperspectral image pairs necessitates the use of dedicated devices (Behmann et al., 2018) and carefully controlled lighting conditions. Additionally, the process incurs significant costs associated with image annotation, while the captured images often exhibit lower resolution. Consequently, acquiring a densely labelled dataset poses considerable challenges. To overcome these limitations, one viable approach is to synthesise RGB and hyperspectral image pairs using a rendering engine and 3D models.

In particular, the Mitsuba engine (Jakob et al., 2022a; Nimier-David et al., 2019; Jakob et al., 2022b) presents capabilities for rendering spectral information and RGB image within a given scene. Leveraging this engine, along with the availability of high-quality 3D indoor models with layout and semantic labels provided by datasets like Structured3d (Zheng et al., 2020) and 3D-FRONT (Fu et al., 2021), and material textures offered by the ambientCG website (Demes), it becomes possible to create a densely labelled material segmentation dataset. By integrating these techniques, a comprehensive pipeline can be established to synthesize the RGB and hyperspectral image pairs, as depicted in Figure 7.2.

Employing this approach not only addresses the challenges associated with data collection but also ensures the availability of densely labelled material segmentation datasets. This synthesised dataset holds great promise for advancing research in the field, facilitating the development and evaluation of material segmentation models in a controlled yet realistic environment.

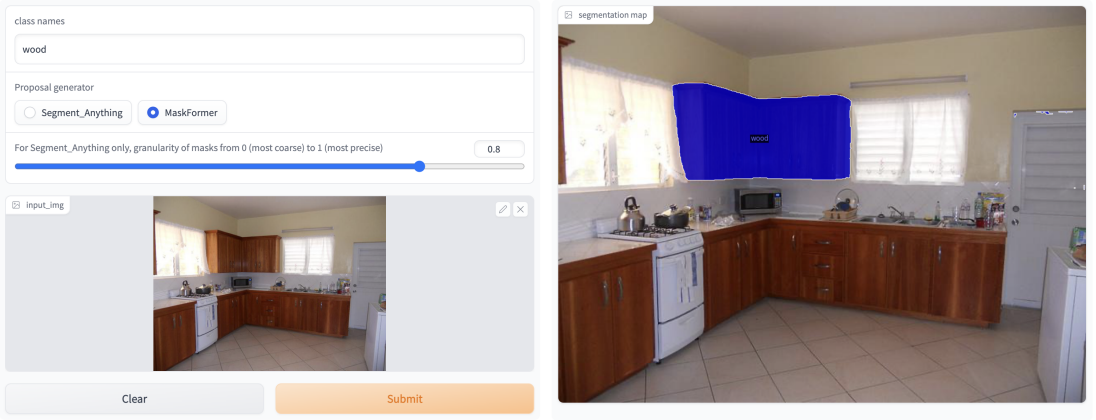


FIGURE 7.3: The example to segment materials with text prompts, using (Liang et al., 2023)

7.6.3 Annotating Sparsely Labelled Datasets

With remarkable advancements in the field of giant models, the acquisition of material segmentations can now be accomplished with minimal human involvement. These powerful models, leveraging the capabilities of deep learning and advanced segmentation techniques, have demonstrated the ability to autonomously infer and delineate segments within images. For example, SAM (Kirillov et al., 2023) can be effectively harnessed to generate material segmentations with enhanced annotations. By integrating additional information in the form of textual descriptions or user clicks that highlight specific material regions of interest, SAM can leverage these inputs to refine and guide the segmentation process.

To incorporate textual descriptions, (Liang et al., 2023) employs a multimodal approach that combines the visual features extracted from the image (by SAM) with the textual embeddings derived from the provided descriptions. By jointly analysing the visual and textual cues, the model can better comprehend the intended material categories and tailor its segmentation outputs accordingly. This integration not only enhances the accuracy of segmentation but also enables SAM to generate more informative and contextually relevant annotations. The illustration is shown in Figure 7.3.

In the case of user clicks, SAM leverages interactive segmentation techniques. Users can interact with the system by clicking on specific regions of the image that correspond to different materials. These clicks serve as valuable guidance for SAM, enabling it to learn from user input and adjust the segmentation boundaries accordingly. By incorporating user clicks, SAM empowers users to actively participate in the annotation process, ensuring that the material segmentations align more closely with their intentions and requirements.

By utilising SAM in conjunction with these additional inputs, the production of material segmentations becomes a collaborative and iterative process. The model learns

FIGURE 7.4: An example to annotate the metal region of the microwave oven with five clicks using (Kirillov et al., 2023).

from the textual descriptions or user clicks to refine its understanding of the material categories and their spatial distributions within the image. As a result, the generated material segmentations exhibit improved accuracy, granularity, and alignment with the intended material boundaries.

This approach not only facilitates the creation of densely labelled material segmentation datasets but also allows for the inclusion of valuable contextual information or user preferences. The integration of textual descriptions or user clicks with the advanced segmentation capabilities of SAM expands the range of possible applications, such as material-based image retrieval, scene understanding, or immersive rendering, where precise material annotations play a crucial role.

By leveraging the capacity of giant models for comprehensive segmentation and incorporating additional guidance, researchers and practitioners can harness its power to generate more accurate and informative material segmentations, ultimately advancing the SOTA in material analysis and related fields.

Appendix A

Extra Analysis and Visualised Images

This section provides more visualised images collected from datasets or produced from experiments to support the argument in this thesis.

A.1 The Limitations of the OpenSurfaces

As mentioned in Section 4.4.1, both of the two material datasets, the LMD (Schwartz and Nishino, 2020, 2016) and the OpenSurfaces (Bell et al., 2013a), are coarsely labelled. For the OpenSurfaces, although Bell et al. (2013a) managed to label the images thoroughly through crowdsourcing, the annotators got confused by the two concepts of material and object. As shown in Figure A.1 and Figure A.2, the materials are segmented by drawing lines along the boundary. Ideally, the lines should be drawn along the boundary between two different materials, such as the ceramic and glass in Figure A.1. However, for the wood and the fabric, the annotators drew the lines alone the object boundary instead of the material boundary. More precisely, in Figure A.1, the table and the floor are made of wood; thus they should be segmented as a whole. In figure A.2, the cushion and the chair are made of fabric, but the segment lines are cautiously kept away from the cushion, and only cover the chair. Segmenting the materials for objects should not be a problem, since material can be considered with a property of object (Farhadi et al., 2009; Zheng et al., 2014). However, for the dense material segmentation task, it is not necessary to isolate the instances of objects, so that the material segments are expected to cross the boundary of objects. The segment lines belonging to object boundary may train the network to identify objects instead of materials. For the LMD, Schwartz and Nishino (2020) intentionally annotated this dataset with segments that cover only a small area of the material region, to avoid the affects of object

boundary, as shown in Figure 3.5. Therefore, we choose to evaluate our DBAT on the LMD rather than the OpenSurfaces.

FIGURE A.1: Annotated image example 1 from the OpenSurfaces.

FIGURE A.2: Annotated image example 2 from the OpenSurfaces.

A.2 Visualisation of sRGB and recovered sRGB pairs

This section presents additional visualisations of pairs of original sRGB images and their corresponding recovered sRGB images. The visualisations reveal interesting insights about different scene conditions. For instance, in the first two rows, where there is no external light source entering the toilet, the recovered images are almost identical to the original images. In contrast, the scenes with windows, as shown in the third, fifth, and sixth rows, demonstrate noticeable variations in the generated heatmaps, emphasising the impact of external lighting on the recovered images. Notably, the image in the fourth row stands out as the sunlight shines on the bed, leading to a significant disparity in the heatmap. These visualisations highlight the need for further refinement of the brightness assumption to regularise the training of the spectral recovery process.

(A) Ground Truth RGB x_m (B) Recovered RGB $\hat{x}_m$ (C) Heatmap of $|x_m - \hat{x}_m|$

FIGURE A.3: The visualisation of more pairs of sRGB and recovered sRGB. The heatmap is measured by averaging the difference between normalised (range [0,1]) x_m and $\hat{x}_m$ across R, G, B channels. Pink means the difference is 0, and red means the difference is 1.

References

Mahmoud Afifi and Michael S Brown. Sensor-independent illumination estimation for dnn models. *arXiv preprint arXiv:1912.06888*, 2019.

Abien Fred Agarap. Deep learning using rectified linear units (relu). *arXiv preprint arXiv:1803.08375*, 2018.

Mirko Agarla, Simone Bianco, Marco Buzzelli, Luigi Celona, and Raimondo Schettini. Fast-n-squeeze: towards real-time spectral reconstruction from rgb images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1132–1139, 2022.

Peri Akiva, Matthew Purri, and Matthew Leotta. Self-supervised material and texture representation learning for remote sensing tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8203–8215, 2022.

Mona Alawadh, Yihong Wu, Yuwen Heng, Luca Remaggi, Mahesan Niranjan, and Hansung Kim. Room acoustic properties estimation from a single 360° photo. In *2022 30th European Signal Processing Conference (EUSIPCO)*, pages 857–861, 2022. .

Daniel R Albert, Michael A Todt, and H Floyd Davis. A low-cost quantitative absorption spectrophotometer. *Journal of Chemical Education*, 89(11):1432–1435, 2012.

Jesus Angulo, Santiago Velasco-Forero, and Jocelyn Chanussot. Multiscale stochastic watershed for unsupervised hyperspectral image segmentation. In *2009 IEEE international geoscience and remote sensing symposium*, volume 3, pages III–93. IEEE, 2009.

Boaz Arad, Radu Timofte, Rony Yahel, Nimrod Morag, Amir Bernat, Yuanhao Cai, Jing Lin, Zudi Lin, Haoqian Wang, Yulun Zhang, et al. Ntire 2022 spectral recovery challenge and data set. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 863–881, 2022.

Vijay Badrinarayanan, Alex Kendall, and Roberto Cipolla. Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 39(12):2481–2495, 2017.

SN Bagayev, VA Averchenko, IA Chekhonin, MA Chekhonin, IM Balmaev, and IB Mekhov. Experimental new ultra-high-speed all-optical coherent streak-camera. In *Journal of Physics: Conference Series*, volume 1695, page 012129. IOP Publishing, 2020.

Dor Bank, Daniel Greenfeld, and Gal Hyams. Improved training for self training by confidence assessments. In *Science and Information Conference*, pages 163–173. Springer, 2018.

SH Shabbeer Basha, Shiv Ram Dubey, Viswanath Pulabaigari, and Snehasis Mukherjee. Impact of fully connected layers on performance of convolutional neural networks for image classification. *Neurocomputing*, 378:112–119, 2020.

David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6541–6549, 2017.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Bolei Zhou, Joshua B Tenenbaum, William T Freeman, and Antonio Torralba. Gan dissection: Visualizing and understanding generative adversarial networks. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

David Bau, Jun-Yan Zhu, Hendrik Strobelt, Agata Lapedriza, Bolei Zhou, and Antonio Torralba. Understanding the role of individual units in a deep neural network. *Proceedings of the National Academy of Sciences*, 117(48):30071–30078, 2020.

Marion F. Baumgardner, Larry L. Biehl, and David A. Landgrebe. 220 band aviris hyperspectral image data set: June 12, 1992 indian pine test site 3, Sep 2015. URL https://purr.purdue.edu/publications/1947/1.

Donald J Baxter and Andrew Murray. Calibration and adaptation of iso visual noise for i3a's camera phone image quality initiative. In *Image Quality and System Performance IX*, volume 8293, pages 21–34. SPIE, 2012.

Jan Behmann, Kelvin Acebron, Dzhaner Emin, Simon Bennertz, Shizue Matsubara, Stefan Thomas, David Bohnenkamp, Matheus T Kuska, Jouni Jussila, Harri Salo, et al. Specim iq: Evaluation of a new, miniaturized handheld hyperspectral camera and its application for plant phenotyping and disease detection. *Sensors*, 18(2):441, 2018.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. OpenSurfaces: A richly annotated catalog of surface appearance. *ACM Trans. on Graphics (SIGGRAPH)*, 32 (4), 2013a.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. bell2013opensurfaces a richly annotated catalog of surface appearance. *ACM Transactions on graphics (TOG)*, 32(4):1–17, 2013b.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. *Computer Vision and Pattern Recognition (CVPR)*, 2015a.

Sean Bell, Paul Upchurch, Noah Snavely, and Kavita Bala. Material recognition in the wild with the materials in context database. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3479–3487, 2015b.

Xavier Berthelon, Guillaume Chenegros, Thomas Finateu, Sio-Hoi Ieng, and Ryad Benosman. Effects of cooling on the snr and contrast detection of a low-light event-based camera. *IEEE transactions on biomedical circuits and systems*, 12(6):1467–1474, 2018.

Alexey Bokhovkin and Evgeny Burnaev. Boundary loss for remote sensing imagery semantic segmentation. In *International Symposium on Neural Networks*, pages 388–401. Springer, 2019.

Shubhankar Borse, Ying Wang, Yizhe Zhang, and Fatih Porikli. Inverseform: A loss function for structured boundary-aware segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5901–5911, 2021.

Sudong Cai, Ryosuke Wakaki, Shohei Nobuhara, and Ko Nishino. Rgb road scene material segmentation. In *Proceedings of the Asian Conference on Computer Vision*, pages 3051–3067, 2022a.

Yuanhao Cai, Jing Lin, Xiaowan Hu, Haoqian Wang, Xin Yuan, Yulun Zhang, Radu Timofte, and Luc Van Gool. Coarse-to-fine sparse transformer for hyperspectral image reconstruction. In *Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XVII*, pages 686–704. Springer, 2022b.

Yuanhao Cai, Jing Lin, Zudi Lin, Haoqian Wang, Yulun Zhang, Hanspeter Pfister, Radu Timofte, and Luc Van Gool. Mst++: Multi-stage spectral-wise transformer for efficient spectral reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 745–755, 2022c.

Hakki Can Karaimer, Iman Khodadad, Farnoud Kazemzadeh, and Michael S Brown. A customized camera imaging pipeline for dermatological imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020.

Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021.

Chih-Hung Chen. Thermal noise measurement and characterization for modern semiconductor devices. *IEEE Instrumentation & Measurement Magazine*, 24(2):60–71, 2021.

Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 357–366, 2021.

Gang Chen, Guipeng Zhang, Zhenguo Yang, and Wenyin Liu. Multi-scale patch-gan with edge detection for image inpainting. *Applied Intelligence*, pages 1–16, 2022.

Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation. In *Proceedings of the European conference on computer vision (ECCV)*, pages 801–818, 2018.

Long Chen, CL Philip Chen, and Mingzhu Lu. A multiple-kernel fuzzy c-means algorithm for image segmentation. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(5):1263–1274, 2011.

Long Chen, Wen Tang, Nigel W John, Tao Ruan Wan, and Jian J Zhang. Context-aware mixed reality: A learning-based framework for semantic-level interaction. In *Computer Graphics Forum*, volume 39, pages 484–496. Wiley Online Library, 2020a.

Wuyang Chen, Ziyu Jiang, Zhangyang Wang, Kexin Cui, and Xiaoning Qian. Collaborative global-local networks for memory-efficient segmentation of ultra-high resolution images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8924–8933, 2019.

Yinpeng Chen, Xiyang Dai, Mengchen Liu, Dongdong Chen, Lu Yuan, and Zicheng Liu. Dynamic convolution: Attention over convolution kernels. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11030–11039, 2020b.

Hao Cheng, Chaochen Gu, and Kaijie Wu. Weakly-supervised semantic segmentation via self-training. In *Journal of Physics: Conference Series*, volume 1487, page 012001. IOP Publishing, 2020.

Norman B. Colthup, Lawrence H. Daly, and Stephen E. Wiberley. Front matter. In *Introduction to Infrared and Raman Spectroscopy (Third Edition)*. Academic Press, San Diego, third edition edition, 1990. ISBN 978-0-12-182554-6. .

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The cityscapes dataset for semantic urban scene understanding. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3213–3223, 2016.

Gabriela Csurka, Diane Larlus, Florent Perronnin, and France Meylan. What is a good evaluation measure for semantic segmentation?. In *BMVC*, volume 27, pages 10–5244, 2013.

Luís Emanuel de Abreu Santos PEREIRA et al. Hyperspectral images applied to the study and conservation of pictorial works. *E-Journal of Portuguese History*, 17(2), 2019.

M.E. Delany and E.N. Bazley. Acoustical properties of fibrous absorbent materials. *Applied Acoustics*, 3(2):105–116, 1970. ISSN 0003-682X. . URL https://www.sciencedirect.com/science/article/pii/0003682X70900319.

Lennart Demes. Public domain resources for physically based rendering. URL https://ambientcg.com/. Accessed: 2021-06-27.

Ilke Demir, Krzysztof Koperski, David Lindenbaum, Guan Pang, Jing Huang, Saikat Basu, Forest Hughes, Devis Tuia, and Ramesh Raskar. Deepglobe 2018: A challenge to parse the earth through satellite images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 172–181, 2018.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009.

Georg Denk and Renate Winkler. Modelling and simulation of transient noise in circuit simulation. *Mathematical and Computer Modelling of Dynamical Systems*, 13(4):383–394, 2007.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2020.

Mark Everingham, Andrew Zisserman, Christopher KI Williams, Luc Van Gool, Moray Allan, Christopher M Bishop, Olivier Chapelle, Navneet Dalal, Thomas Deselaers, Gyuri Dorkó, et al. The 2005 pascal visual object classes challenge. In *Machine Learning Challenges Workshop*, pages 117–176. Springer, 2005.

Leon Eversberg and Jens Lambrecht. Generating images with physics-based rendering for an industrial object detection task: Realism versus domain randomization. *Sensors*, 21(23):7901, 2021.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1785. IEEE, 2009.

Roland W Fleming. Visual perception of materials and their properties. *Vision Research*, 94:62–75, 2014. ISSN 0042-6989. .

Huan Fu, Bowen Cai, Lin Gao, Ling-Xiao Zhang, Jiaming Wang, Cao Li, Qixun Zeng, Chengyue Sun, Rongfei Jia, Binqiang Zhao, et al. 3d-front: 3d furnished rooms with layouts and semantics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10933–10942, 2021.

Hiroshi Fukui, Tsubasa Hirakawa, Takayoshi Yamashita, and Hironobu Fujiyoshi. Attention branch network: Learning of attention mechanism for visual explanation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10705–10714, 2019.

Rong Ge, Sham M Kakade, Rahul Kidambi, and Praneeth Netrapalli. The step decay schedule: A near optimal, geometrically decaying learning rate procedure for least squares. *Advances in Neural Information Processing Systems*, 32, 2019.

Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bygh9j09KX.

Pedram Ghamisi, Micael S Couceiro, Jón Atli Benediktsson, and Nuno MF Ferreira. An efficient method for segmentation of images based on fractional calculus and natural selection. *Expert Systems with Applications*, 39(16):12407–12417, 2012.

Pedram Ghamisi, Micael S Couceiro, Fernando ML Martins, and Jon Atli Benediktsson. Multilevel image segmentation based on fractional-order darwinian particle swarm optimization. *IEEE Transactions on Geoscience and Remote sensing*, 52(5):2382–2394, 2013.

Golnaz Ghiasi and Charless C Fowlkes. Laplacian pyramid reconstruction and refinement for semantic segmentation. In *European conference on computer vision*, pages 519–534. Springer, 2016.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

Akhilesh Gotmare, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. A closer look at deep learning heuristics: Learning rate restarts, warmup and distillation. In *International Conference on Learning Representations*, 2018.

Reaya Grewal, Singara Singh Kasana, and Geeta Kasana. Hyperspectral image segmentation: a comprehensive survey. *Multimedia Tools and Applications*, pages 1–54, 2022.

Reaya Grewal, Singara Singh Kasana, and Geeta Kasana. Hyperspectral image segmentation: a comprehensive survey. *Multimedia Tools and Applications*, 82(14):20819–20872, 2023.

Baoshan Guo, Jingya Sun, YongFeng Lu, and Lan Jiang. Ultrafast dynamics observation during femtosecond laser-material interaction. *International Journal of Extreme Manufacturing*, 1(3):032004, 2019.

Dianyuan Han. Comparison of commonly used image interpolation methods. In *Conference of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, pages 1556–1559. Atlantis Press, 2013.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Enhancing material features using dynamic backward attention on cross-resolution patches. In *33rd British Machine Vision Conference 2022, BMVC 2022, London, UK, November 21-24, 2022*. BMVA Press, 2022a.

Yuwen Heng, Yihong Wu, Srinandan Dasmahapatra, and Hansung Kim. Cam-segnet: A context-aware dense material segmentation network for sparsely labelled datasets. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISIGRAPP 2022) - Volume 5: VISAPP*, pages 190–201. INSTICC, SciTePress, 2022b. ISBN 978-989-758-555-5. .

Yuwen Heng, Srinandan Dasmahapatra, and Hansung Kim. Material recognition for immersive interactions in virtual/augmented reality. In *2023 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 577–578, 2023. .

Tomáš Hodaň, Vibhav Vineet, Ran Gal, Emanuel Shalev, Jon Hanzelka, Treb Connell, Pedro Urbina, Sudipta N Sinha, and Brian Guenter. Photorealistic image synthesis for object instance detection. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 66–70. IEEE, 2019.

Alexander Horn. *Ultra-fast material metrology*. John Wiley & Sons, 2009.

Xiaowan Hu, Yuanhao Cai, Jing Lin, Haoqian Wang, Xin Yuan, Yulun Zhang, Radu Timofte, and Luc Van Gool. Hdnet: High-resolution dual-domain learning for spectral compressive imaging. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17542–17551, 2022.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q Weinberger. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4700–4708, 2017.

Dongseok Im, Donghyeon Han, Sungpill Choi, Sanghoon Kang, and Hoi-Jun Yoo. Dt-cnn: Dilated and transposed convolution neural network accelerator for real-time image segmentation on mobile devices. In *2019 IEEE international symposium on circuits and systems (ISCAS)*, pages 1–5. IEEE, 2019.

Sara Iodice and Krystian Mikolajczyk. Text attribute aggregation and visual feature decomposition for person search. In *BMVC*, 2020.

Chandni Jain, Akshay Bhargava, Sharad Gupta, Rishi Rath, Abhishek Nagpal, and Prince Kumar. Spectrophotometric evaluation of the color changes of different feldspathic porcelains after exposure to commonly consumed beverages. *European Journal of Dentistry*, 7(02):172–180, 2013.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, and Delio Vicini. Dr. jit: a just-in-time compiler for differentiable rendering. *ACM Transactions on Graphics (TOG)*, 41 (4):1–19, 2022a.

Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022b. https://mitsuba-renderer.org.

J Alstan Jakubiec. Building a database of opaque materials for lighting simulation. In *PLEA 2016–Cities, Buildings, People: Towards Regenerative Environments,). Proceedings of the 32nd International Conference on Passive and Low Energy Architecture.*, 2016.

J Alstan Jakubiec. Data-driven selection of typical opaque material reflectances for lighting simulation. *LEUKOS*, pages 1–14, 2022.

Yuhyun Ji, Yunsang Kwak, Sang Mok Park, and Young L Kim. Compressive recovery of smartphone rgb spectral sensitivity functions. *Optics Express*, 29(8):11947–11961, 2021.

Yan Jia, Yinqiang Zheng, Lin Gu, Art Subpa-Asa, Antony Lam, Yoichi Sato, and Imari Sato. From rgb to spectrum for natural scenes via manifold-based mapping. In *Proceedings of the IEEE international conference on computer vision*, pages 4705–4713, 2017.

Nathaniel L Jones and Christoph F Reinhart. Experimental validation of ray tracing as a means of image-based visual discomfort prediction. *Building and Environment*, 113: 131–150, 2017.

Linda S Kalman and Edward M Bassett III. Classification and material identification in an urban environment using hydice hyperspectral data. In *Imaging Spectrometry III*, volume 3118, pages 57–68. SPIE, 1997.

Jian Kang, Ruben Fernandez-Beltran, Xian Sun, Jingen Ni, and Antonio Plaza. Deep learning-based building footprint extraction with missing annotations. *IEEE Geoscience and Remote Sensing Letters*, 2021.

Hansung Kim, Luca Remaggi, Philip JB Jackson, and Adrian Hilton. Immersive spatial audio reproduction for vr/ar using room acoustic modelling from 360 images. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 120–126. IEEE, 2019.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

David R Kirkby and David T Delpy. Measurement of tissue temporal point spread function (tpsf) by use of a gain-modulated avalanche photodiode detector. *Physics in Medicine & Biology*, 41(5):939, 1996.

Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

Philipp Krähenbühl and Vladlen Koltun. Parameter learning and convergent inference for dense random fields. In *International Conference on Machine Learning*, pages 513–521. PMLR, 2013.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 25. Curran Associates, Inc., 2012.

Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

T Hoang Ngan Le, Khoa Luu, and Marios Savvides. Fast and robust self-training beard/moustache detection and segmentation. In *2015 international conference on biometrics (ICB)*, pages 507–512. IEEE, 2015.

Jiangbo Li, Liping Chen, and Wenqian Huang. Detection of early bruises on peaches (amygdalus persica l.) using hyperspectral imaging coupled with improved watershed segmentation algorithm. *Postharvest Biology and Technology*, 135:104–113, 2018.

Jiangbo Li, Wei Luo, Zheli Wang, and Shuxiang Fan. Early detection of decay on apples using hyperspectral reflectance imaging combining both principal component analysis and improved watershed segmentation method. *Postharvest Biology and Technology*, 149:235–246, 2019.

Jiaojiao Li, Songcheng Du, Chaoxiong Wu, Yihong Leng, Rui Song, and Yunsong Li. Drcr net: Dense residual channel re-calibration network with non-local purification for spectral super resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1259–1268, 2022a.

Mingsong Li, Yikun Liu, Guangkuo Xue, Yuwen Huang, and Gongping Yang. Exploring the relationship between center and neighborhoods: Central vector oriented self-similarity network for hyperspectral image classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2022b.

Mingsong Li, Wei Li, Yikun Liu, Yuwen Huang, and Gongping Yang. Adaptive mask sampling and manifold to euclidean subspace learning with distance covariance representation for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 2023.

Zhengying Li, Hong Huang, and Zhen Zhang. Deep manifold reconstruction neural network for hyperspectral image classification. *IEEE Geoscience and Remote Sensing Letters*, 19:1–5, 2020.

Feng Liang, Bichen Wu, Xiaoliang Dai, Kunpeng Li, Yinan Zhao, Hang Zhang, Peizhao Zhang, Peter Vajda, and Diana Marculescu. Open-vocabulary semantic segmentation with mask-adapted clip. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7061–7070, 2023.

Yupeng Liang, Ryosuke Wakaki, Shohei Nobuhara, and Ko Nishino. Multimodal material segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19800–19808, 2022.

Jeff W Lichtman and José-Angel Conchello. Fluorescence microscopy. *Nature methods*, 2(12):910–919, 2005.

Jyh-Woei Lin. Artificial neural network related to biological neuron network: a review. *Advanced Studies in Medical Sciences*, 5(1):55–62, 2017.

Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017a.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017b.

Fenglin Liu, Xuancheng Ren, Zhiyuan Zhang, Xu Sun, and Yuexian Zou. Rethinking skip connection with layer normalization. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3586–3598, 2020.

Jiaying Liu, Shuai Yang, Yuming Fang, and Zongming Guo. Structure-guided image inpainting using homography transformation. *IEEE Transactions on Multimedia*, 20 (12):3252–3265, 2018a.

Nian Liu, Junwei Han, and Ming-Hsuan Yang. Picanet: Learning pixel-wise contextual attention for saliency detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3089–3098, 2018b.

Sicong Liu, Daniele Marinelli, Lorenzo Bruzzone, and Francesca Bovolo. A review of change detection in multitemporal hyperspectral images: Current techniques, applications, and challenges. *IEEE Geoscience and Remote Sensing Magazine*, 7(2):140–158, 2019a.

Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. *arXiv preprint arXiv:2111.09883*, 2021a.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021b.

Zhiqiang Liu, Paul Chow, Jinwei Xu, Jingfei Jiang, Yong Dou, and Jie Zhou. A uniform architecture design for accelerating 2d and 3d cnns on fpgas. *Electronics*, 8(1):65, 2019b.

Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

Robert J Lucas, Stuart N Peirson, David M Berson, Timothy M Brown, Howard M Cooper, Charles A Czeisler, Mariana G Figueiro, Paul D Gamlin, Steven W Lockley, John B O'Hagan, et al. Measuring and using light in the melanopsin age. *Trends in neurosciences*, 37(1):1–9, 2014.

Holger Ludvigsen and Anne C Elster. Real-time ray tracing using nvidia optix. In *Eurographics (Short Papers)*, pages 65–68, 2010.

François Lureau, Guillaume Matras, Olivier Chalus, Christophe Derycke, Thomas Morbieu, Christophe Radier, Olivier Casagrande, Sébastien Laux, Sandrine Ricaud, Gilles

Rey, et al. High-energy hybrid femtosecond laser system demonstrating 2× 10 pw capability. *High Power Laser Science and Engineering*, 8:e43, 2020.

Hualiang Lv, Zhihong Yang, Hongge Pan, and Renbing Wu. Electromagnetic absorption materials: Current progress and new frontiers. *Progress in Materials Science*, page 100946, 2022.

Jerry Ma and Denis Yarats. On the adequacy of untuned warmup for adaptive optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 8828–8836, 2021.

Magnus Magnusson, Jakob Sigurdsson, Sveinn Eirikur Armansson, Magnus O Ulfarsson, Hilda Deborah, and Johannes R Sveinsson. Creating rgb images from hyperspectral images using a color matching function. In *IGARSS 2020-2020 IEEE International Geoscience and Remote Sensing Symposium*, pages 2045–2048. IEEE, 2020.

Shi Mao, Mengqi Ji, Bin Wang, Qionghai Dai, and Lu Fang. Surface material perception through multimodal learning. *IEEE Journal of Selected Topics in Signal Processing*, 16 (4):843–853, 2022.

Miguel Ángel Martínez-Domingo, Eva M Valero, Javier Hernández-Andrés, Shoji Tominaga, Takahiko Horiuchi, and Keita Hirai. Image processing pipeline for segmentation and material classification based on multispectral high dynamic range polarimetric images. *Optics express*, 25(24):30073–30090, 2017.

Aoife McDonagh, Joseph Lemley, Ryan Cassidy, and Peter Corcoran. Synthesizing game audio using deep neural networks. In *2018 IEEE Games, Entertainment, Media Conference (GEM)*, pages 1–9. IEEE, 2018.

Aditya Mehta, Harsh Sinha, Murari Mandal, and Pratik Narang. Domain-aware unsupervised hyperspectral reconstruction for aerial image dehazing. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pages 413–422, 2021.

Daniel Meister, Shinji Ogaki, Carsten Benthin, Michael J Doyle, Michael Guthe, and Jiří Bittner. A survey on bounding volume hierarchies for ray tracing. In *Computer Graphics Forum*, volume 40, pages 683–712. Wiley Online Library, 2021.

Andres Milioto, Ignacio Vizzo, Jens Behley, and Cyrill Stachniss. Rangenet++: Fast and accurate lidar semantic segmentation. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 4213–4220. IEEE, 2019.

Seyedali Mirjalili and Seyedali Mirjalili. Particle swarm optimisation. *Evolutionary Algorithms and Neural Networks: Theory and Applications*, pages 15–31, 2019.

Dipti Mishra, Satish Kumar Singh, and Rajat Kumar Singh. Deep architectures for image compression: a critical review. *Signal Processing*, 191:108346, 2022.

Purnendu Mishra and Kishor Sarawadekar. Polynomial learning rate policy with warm restart for deep neural network. In *TENCON 2019-2019 IEEE Region 10 Conference (TENCON)*, pages 2087–2092. IEEE, 2019.

Yujian Mo, Yan Wu, Xinneng Yang, Feilin Liu, and Yujun Liao. Review the state-of-the-art technologies of semantic segmentation based on deep learning. *Neurocomputing*, 493:626–646, 2022.

L. Mu. Efficient mini-batch training for stochastic optimization. *ACM*, 2014.

Thao Nguyen, Maithra Raghu, and Simon Kornblith. Do wide and deep networks learn the same things? uncovering how neural network representations vary with width and depth. *International Conference on Learning Representations*, 2020.

Merlin Nimier-David, Delio Vicini, Tizian Zeltner, and Wenzel Jakob. Mitsuba 2: A retargetable forward and inverse renderer. *ACM Transactions on Graphics (TOG)*, 38 (6):1–17, 2019.

Chan Rok Park, Seong-Hyeon Kang, and Youngjin Lee. Median modified wiener filter for improving the image quality of gamma camera images. *Nuclear Engineering and Technology*, 52(10):2328–2333, 2020.

Matt Pharr, Wenzel Jakob, and Greg Humphreys. *Physically based rendering: From theory to implementation*. Morgan Kaufmann, 2016.

Rodrigo Jose Pisani, Rodrigo Yuji Mizobe Nakamura, Paulina Setti Riedel, Célia Regina Lopes Zimback, Alexandre Xavier Falcao, and João Paulo Papa. Toward satellite-based land cover classification through optimum-path forest. *IEEE Transactions on Geoscience and Remote Sensing*, 52(10):6075–6085, 2014.

D Chandra Prakash, RC Narayanan, N Ganesh, M Ramachandran, S Chinnasami, and R Rajeshwari. A study on image processing with data analysis. In *AIP Conference Proceedings*, volume 2393, page 020225. AIP Publishing LLC, 2022.

Yuhao Qing, Wenyi Liu, Liuyan Feng, and Wanjia Gao. Improved transformer net for hyperspectral image classification. *Remote Sensing*, 13(11):2216, 2021.

Florian Quatresooz, Simon Demey, and Claude Oestges. Tracking of interaction points for improved dynamic ray tracing. *IEEE Transactions on Vehicular Technology*, 70(7): 6291–6301, 2021.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021.

Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34, 2021.

Md Atiqur Rahman and Yang Wang. Optimizing intersection-over-union in deep neural networks for image segmentation. In *International symposium on visual computing*, pages 234–244. Springer, 2016.

Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

Relindis Rott. Dynamic update of stand-alone lidar model based on ray tracing using the nvidia optix engine. In *2022 International Conference on Connected Vehicle and Expo (ICCVE)*, pages 1–6. IEEE, 2022.

Stéphane Roussel, Matthieu Boffety, and François Goudail. Polarimetric precision of micropolarizer grid-based camera in the presence of additive and poisson shot noise. *Optics express*, 26(23):29968–29982, 2018.

Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? *Advances in neural information processing systems*, 31, 2018.

Vishwanath Saragadam and Aswin C Sankaranarayanan. Programmable spectrometry: per-pixel material classification using learned spectral filters. In *2020 IEEE International Conference on Computational Photography (ICCP)*, pages 1–10. IEEE, 2020.

Gabriel Schwartz. *Visual Material Recognition*. Drexel University, 2018.

Gabriel Schwartz and Ko Nishino. Visual material traits: Recognizing per-pixel material context. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 883–890, 2013.

Gabriel Schwartz and Ko Nishino. Material recognition from local appearance in global context. In *Biol. and Artificial Vision (Workshop held in conjunction with ECCV 2016)*, 2016.

Gabriel Schwartz and Ko Nishino. Recognizing material properties from images. *IEEE transactions on pattern analysis and machine intelligence*, 42(8):1981–1995, 2019.

Gabriel Schwartz and Ko Nishino. Recognizing material properties from images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(8):1981–1995, 2020. .

Muhammad Saad Shaikh, Keyvan Jaferzadeh, Benny Thörnberg, and Johan Casselgren. Calibration of a hyper-spectral imaging system using a low-cost reference. *Sensors*, 21(11):3738, 2021.

Wen Shen, Zhihua Wei, Shikun Huang, Binbin Zhang, Jiaqi Fan, Ping Zhao, and Quanshi Zhang. Interpretable compositional convolutional neural networks. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 2021.

Yuanyuan Shen, Edmund M-K Lai, and Mahsa Mohaghegh. Effects of similarity score functions in attention mechanisms on the performance of neural question answering systems. *Neural Processing Letters*, pages 1–20, 2022.

Richard Shin and Dawn Song. Jpeg-resistant adversarial images. In *NIPS 2017 Workshop on Machine Learning and Computer Security*, volume 1, page 8, 2017.

Ukcheol Shin, Jinsun Park, Gyumin Shim, Francois Rameau, and In So Kweon. Camera exposure control for robust robot vision with noise-aware image quality assessment. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1165–1172. IEEE, 2019.

Nithin Shrivatsav, Lakshmi Nair, and Sonia Chernova. Tool substitution with shape and material reasoning using dual neural networks. *arXiv preprint arXiv:1911.04521*, 2019.

Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *European conference on computer vision*, pages 746–760. Springer, 2012.

Leslie N Smith. Cyclical learning rates for training neural networks. In *2017 IEEE winter conference on applications of computer vision (WACV)*, pages 464–472. IEEE, 2017.

PR Smith. Bilinear interpolation of digital images. *Ultramicroscopy*, 6(2):201–204, 1981.

Vivianne C Smith and Joel Pokorny. The design and use of a cone chromaticity space: a tutorial. *Color Research & Application*, 21(5):375–383, 1996.

Hannah E Smithson. Sensory, computational and cognitive components of human colour constancy. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 360(1458):1329–1346, 2005.

Jifei Song, Yongxin Yang, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Generalizable person re-identification by domain-invariant mapping network. In *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pages 719–728, 2019.

Le Song, Alex Smola, Arthur Gretton, Justin Bedo, and Karsten Borgwardt. Feature selection via dependence maximization. *Journal of Machine Learning Research*, 13(5), 2012.

Nasim Souly, Concetto Spampinato, and Mubarak Shah. Semi supervised semantic segmentation using generative adversarial network. In *Proceedings of the IEEE international conference on computer vision*, pages 5688–5696, 2017.

Robin Strudel, Ricardo Garcia, Ivan Laptev, and Cordelia Schmid. Segmenter: Transformer for semantic segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7262–7272, 2021.

Mary B Stuart, Matthew Davies, Matthew J Hobbs, Tom D Pering, Andrew JS McGonigle, and Jon R Willmott. High-resolution hyperspectral imaging using low-cost components: Application within environmental monitoring scenarios. *Sensors*, 22 (12):4652, 2022.

Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik Learned-Miller, and Jan Kautz. Pixel-adaptive convolutional neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11166–11175, 2019.

Shuochen Su, Felix Heide, Robin Swanson, Jonathan Klein, Clara Callenberg, Matthias Hullin, and Wolfgang Heidrich. Material classification using raw time-of-flight measurements. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3503–3511, 2016.

Linli Sun and Feng Zhao. Bidirectional reflectance distribution function algorithm based on the poynting vector analysis. *Optical Engineering*, 60(6):063104, 2021.

Charles Sutton and Andrew McCallum. An introduction to conditional random fields for relational learning. *Introduction to statistical relational learning*, 2:93–128, 2006.

Mingxing Tan and Quoc Le. Efficientnet: Rethinking model scaling for convolutional neural networks. In *International conference on machine learning*, pages 6105–6114. PMLR, 2019.

Zhenyu Tang, Nicholas J Bryan, Dingzeyu Li, Timothy R Langlois, and Dinesh Manocha. Scene-aware audio rendering via deep acoustic analysis. *IEEE transactions on visualization and computer graphics*, 26(5):1991–2001, 2020.

Andrew Tao, Karan Sapra, and Bryan Catanzaro. Hierarchical multi-scale attention for semantic segmentation. *arXiv preprint arXiv:2005.10821*, 2020.

Marvin Teichmann and R. Cipolla. Convolutional crfs for semantic segmentation. In *BMVC*, 2019.

Ilya O Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *Advances in Neural Information Processing Systems*, 34, 2021.

Lisa Torrey and Jude Shavlik. Transfer learning. In *Handbook of research on machine learning applications and trends: algorithms, methods, and techniques*, pages 242–264. IGI global, 2010.

Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

Joel A Tropp. A mathematical introduction to compressive sensing [book review]. *Bulletin of the American Mathematical Society*, 54(1):151–165, 2017.

Ethan Tseng, Yuxuan Zhang, Lars Jebe, Xuaner Zhang, Zhihao Xia, Yifei Fan, Felix Heide, and Jiawen Chen. Neural photo-finishing. *ACM Transactions on Graphics (TOG)*, 41(6):1–15, 2022.

Bing Tu, Wangquan He, Wei He, Xianfeng Ou, and Antonio Plaza. Hyperspectral classification via global-local hierarchical weighting fusion network. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 15:184–200, 2021.

Peter A van Nijnatten. Regular reflectance and transmittance. In *Experimental Methods in the Physical Sciences*, volume 46, pages 143–178. Elsevier, 2014.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Hanuman Verma, RK Agrawal, and Aditi Sharan. An improved intuitionistic fuzzy c-means clustering algorithm incorporating local information for brain image segmentation. *Applied Soft Computing*, 46:543–557, 2016.

Dezhao Wang, Wenhan Yang, Yueyu Hu, and Jiaying Liu. Neural data-dependent transform for learned image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17379–17388, 2022.

Zijie J Wang, Robert Turko, Omar Shaikh, Haekyu Park, Nilaksh Das, Fred Hohman, Minsuk Kahng, and Duen Horng Polo Chau. Cnn explainer: Learning convolutional neural networks with interactive visualization. *IEEE Transactions on Visualization and Computer Graphics*, 27(2):1396–1406, 2020.

Yunchao Wei, Huaxin Xiao, Honghui Shi, Zequn Jie, Jiashi Feng, and Thomas S Huang. Revisiting dilated convolution: A simple approach for weakly-and semi-supervised semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7268–7277, 2018.

Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. The marginal value of adaptive gradient methods in machine learning. *Advances in neural information processing systems*, 30, 2017.

Tong Wu, Zhenzhen Lei, Bingqian Lin, Cuihua Li, Yanyun Qu, and Yuan Xie. Patch proposal network for fast semantic segmentation of high-resolution images. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 12402–12409, 2020.

Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation from a single omnidirectional image using domain adaptation. In *European Conference on Visual Media Production (CVMP)*, pages 1–9, 2021.

Yihong Wu, Yuwen Heng, Mahesan Niranjan, and Hansung Kim. Depth estimation for a single omnidirectional image with reversed-gradient warming-up thresholds discriminator. In *2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2023.

Yinhua Wu, Bingliang Hu, Xiaohui Gao, and Ruyi Wei. Hyperspectral image classification based on adaptive segmentation. *Optik*, 172:612–621, 2018.

Junshi Xia, Lionel Bombrun, Tülay Adalı, Yannick Berthoumieu, and Christian Germain. Spectral–spatial classification of hyperspectral images using ica and edge-preserving filter via an ensemble strategy. *IEEE Transactions on Geoscience and Remote Sensing*, 54(8):4971–4982, 2016.

Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

Jianxiong Xiao, Krista A Ehinger, James Hays, Antonio Torralba, and Aude Oliva. Sun database: Exploring a large collection of scene categories. *International Journal of Computer Vision*, 119(1):3–22, 2016.

Mingzhou Xu, Derek F. Wong, Baosong Yang, Yue Zhang, and Lidia S. Chao. Leveraging local and global patterns for self-attention networks. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3069–3075, Florence, Italy, jul 2019. Association for Computational Linguistics.

Zhaohui Xue, Mengxue Zhang, Yifeng Liu, and Peijun Du. Attention-based second-order pooling network for hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 59(11):9600–9615, 2021.

Pavel Yakubovskiy. Segmentation models pytorch. `https://github.com/qubvel/segmentation_models.pytorch`, 2020.

Nagaraj Yamanakkanavar and Bumshik Lee. Using a patch-wise m-net convolutional neural network for tissue segmentation in brain mri images. *IEEE Access*, 8:120946–120958, 2020.

Sungwook Youn and Chulhee Lee. Edge detection for hyperspectral images using the bhattacharyya distance. In *2013 International Conference on Parallel and Distributed Systems*, pages 716–719. IEEE, 2013.

Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *arXiv preprint arXiv:2004.02147*, 2020.

Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *International Journal of Computer Vision*, 129:3051–3068, 2021a.

Fei Yu, Jiji Tang, Weichong Yin, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. Ernie-vil: Knowledge enhanced vision-language representations through scene graphs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 3208–3216, 2021b.

Chao Zhang, Jérémy Marty, Anne Maynadier, Philippe Chaudet, Julien Réthoré, and Marie-Christine Baietto. An innovative technique for real-time adjusting exposure time of silicon-based camera to get stable gray level images with temperature evolution. *Mechanical Systems and Signal Processing*, 122:419–432, 2019a.

Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Zhi Zhang, Haibin Lin, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. *arXiv preprint arXiv:2004.08955*, 2020a.

Hang Zhang, Chongruo Wu, Zhongyue Zhang, Yi Zhu, Haibin Lin, Zhi Zhang, Yue Sun, Tong He, Jonas Mueller, R Manmatha, et al. Resnest: Split-attention networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2736–2746, 2022.

Hongyan Zhang, Yue Liao, Honghai Yang, Guangyi Yang, and Liangpei Zhang. A local-global dual-stream network for building extraction from very-high-resolution remote sensing images. *IEEE Transactions on Neural Networks and Learning Systems*, 2020b.

Hongyang Zhang, Junru Shao, and Ruslan Salakhutdinov. Deep neural networks with multi-branch architectures are intrinsically less non-convex. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 1099–1109. PMLR, 2019b.

Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018a.

Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8827–8836, 2018b.

Cheng Zhao, Li Sun, and Rustam Stolkin. A fully end-to-end deep learning approach for real-time simultaneous 3d reconstruction and material recognition. In *2017 18th International Conference on Advanced Robotics (ICAR)*, pages 75–82. IEEE, 2017a.

Cheng Zhao, Li Sun, and Rustam Stolkin. Simultaneous material segmentation and 3d reconstruction in industrial scenarios. *Frontiers in Robotics and AI*, 7:52, 2020a.

Hengshuang Zhao, Jianping Shi, Xiaojuan Qi, Xiaogang Wang, and Jiaya Jia. Pyramid scene parsing network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2881–2890, 2017b.

Hengshuang Zhao, Jiaya Jia, and Vladlen Koltun. Exploring self-attention for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10076–10085, 2020b.

Jia Zheng, Junfei Zhang, Jing Li, Rui Tang, Shenghua Gao, and Zihan Zhou. Structured3d: A large photo-realistic dataset for structured 3d modeling. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IX 16*, pages 519–535. Springer, 2020.

Shuai Zheng, Ming-Ming Cheng, Jonathan Warrell, Paul Sturgess, Vibhav Vineet, Carsten Rother, and Philip HS Torr. Dense semantic image segmentation with objects and attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3214–3221, 2014.

Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. Conditional random fields as recurrent neural networks. In *Proceedings of the IEEE international conference on computer vision*, pages 1529–1537, 2015.

Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6881–6890, 2021.

Yanfei Zhong, Xinyu Wang, Lin Zhao, Ruyi Feng, Liangpei Zhang, and Yanyan Xu. Blind spectral unmixing based on sparse component analysis for hyperspectral remote sensing imagery. *ISPRS Journal of Photogrammetry and Remote Sensing*, 119:49–63, 2016.

Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 633–641, 2017.

Bolei Zhou, David Bau, Aude Oliva, and Antonio Torralba. Interpreting deep visual representations via network dissection. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2131–2145, 2018.

Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Semantic understanding of scenes through the ade20k dataset. *International Journal of Computer Vision*, 127(3):302–321, 2019a.

Feng Zhou, Renlong Hang, Qingshan Liu, and Xiaotong Yuan. Hyperspectral image classification using spectral-spatial lstms. *Neurocomputing*, 328:39–47, 2019b.

Xiaojin Jerry Zhu. Semi-supervised learning literature survey. 2005.

Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. *arXiv preprint arXiv:2011.10033*, 2020.

Barret Zoph, Golnaz Ghiasi, Tsung-Yi Lin, Yin Cui, Hanxiao Liu, Ekin Dogus Cubuk, and Quoc Le. Rethinking pre-training and self-training. In *NeurIPS*, 2020.