Extraction and Description of 3D (Articulated) Moving Objects

Karl J Sharman, Mark S Nixon and John N Carter
ISIS Research Group, Department of Electronics and Computer Science
University of Southampton, UK.
{kjs98r, msn, jnc}@ecs.soton.ac.uk

Abstract

We seek to extract the three-dimensional motion of articulated objects by passive sensing. We first developed a multi-view system that segments objects from the background before their subsequent analysis by a motion model. To reduce the effects of noise, we also performed 3D reconstruction prior to motion analysis to improve the extraction of the (static) background. The 3D extraction and motion analysis are both performed by evidence gathering, thus accruing optimality of performance. These new analyses are supported by a new representation which describes 3D data with optimal fidelity.

1. Introduction

The goal for our systems is the gathering of information, without contact, from abstract arbitrary scenes, with a mathematical model of the object being sought as the only *a priori* information permitted. One further requirement is the ability to handle noise, however, there are actually three sources: those due to the initial image capture, those due to other objects in the scene, and those inherent in the viewed object: human gait is inherently self-occluding.

An increasingly common choice for the construction of 3D data is the use of multiple views. Two options exist which are to analyse the 2D image sequences with a 3D model or to analyse 3D data, created from the 2D images, with a 3D model.

The first option removes the background and any movement found would be passed directly to the evidence gathering algorithm which calculates the best fitness for the dynamic model in the scene. The model is temporal and 3D in nature, but would be mapped onto the 2D static images.

The second option requires a method to reconstruct 3D objects from the 2D images. Earlier work by Martin *et al* [3] demonstrated a method called Volume Intersection (VI) which was capable of combining multiple segmented views to generate a 3D description of the object. However, concavities cannot be resolved with this algorithm, and a feature known as the visual hull has been defined by Laurentini [2]. The VI algorithm is a simple yet effective method for the capture of 3D data. With the initial segmentation performed correctly, the object is known to exist within the resulting visual hull; an exact model can be rendered for objects with no concavities.

Recently, efforts have been made to produce colour and grey-scale algorithms based upon VI, endeavouring to produce better-than-hull results by conferring shade information about the concavities thereby obtaining a better approximation of the underlying object. This research stems from work by Seitz *et al* [6], called voxel coloring, where the construction of the scene is by depth order of voxels in the resulting voxel space. Each voxel is determined to be either coloured or transparent. If it is coloured, it will have occluding properties on later voting, hence the need to perform the sweep of the voxel space in depth order.

Thus the second option would construct the 3D model on a frame-by-frame basis using similar reconstruction techniques. Analysis would be performed using evidence gathering techniques, as demonstrated by Cunado *et al* [1] and Nash *et al* [5]. The dynamic information in this 3D scene would be extracted by removal of the background, and the evidence gathering procedure would then be used to parameterise the information. However, in addition to a voxel-based approach for this 3D representation, we propose a new 2.75D full colour data representation, and thus three implementations will be described and will henceforth be called the 2D, 3D and 2.75D systems.

2. 2D system background removal

For a specific pixel in all of the images from the same field, medians and standard deviations (σ) of the three colour components of the source images are sought. The median is used as an approximation of the mode, and is thus the estimated background colour. Pixels are deemed to be background if all three colour components lie within a distance of max$(k_1\sigma, k_2)$ of the respective median, where

k_1 and k_2 are currently selected experimentally. The use of the standard deviation enables regions of high disturbance to be removed.

3. 3D system data generation

The 3D system must generate a 3D scene on a frame-by-frame basis. This is performed using a similar method to voxel coloring described by Seitz *et al* [6], however, the statistical nature of our approach does not require the voxel space to be swept in a particular direction in order to take into account occlusion: for each voxel, its shade, and the confidence in the shade is calculated from the rays which pass through it, based upon a statistical measure m, where,

$$m = \frac{\sigma^2 + k_1}{n + k_2} \tag{1}$$

where σ^2 is the variance of the grey level of the n contributing rays. k_1 and k_2 adjust the weight of voting for more views. Only the voxels that indicate the highest confidence are selected; these will cause occlusions for the further iterations. During the processing, voxels are allocated six sides thus facing views cannot vote against each other.

Once a sequence of voxel spaces has been created, background voxels are removed in a manner similar to that used in the 2D system, noting that there is a special case of transparency before the background is removed. An alternative method is to mask out voxels that are not in the visual hull formed from the 2D segmented images, hence producing *better-than-hull* results.

4. 2.75D system data generation

Voxels are an inherently poor approach to 3D reconstruction. For a regular spaced voxel grid, voxels near to the camera's view may cover a large number of pixels. For an object in the foreground, this means that information about it is being discarded as the contributing pixels are merged into a single voxel. For distant objects, there is the possibility of voxels oversampling it.

Extending the definition of 2.5D images, where each pixel has an associated depth, in our new set of algorithms, each pixel may have many associated depths. This increases their flexibility to represent data similar to that obtained by the previously described grey voxel-based reconstruction algorithm. We call this new representation the 2.75D image. On their own, 2.5D and 2.75D images cannot fully describe the 3D world; only by combining the multiple views with a union operation can this be achieved.

With this new representation, each pixel is projected, i.e., ray-cast, and compared with all the possible combinations from other views. By considering the manner in which the rays are projected onto the other images, the rays can be projected at an optimum rate through this boundless and near-infinite space. This is related to the work by Matusik *et al* [4], however, our colour algorithm does not require the object to be segmented from the background.

4.1 Implementing VI

The method by which information is gathered and reconstructed using VI in 2.75D will now be considered.

Given n cameras that view the subject ($\subset \mathbb{R}^3$), each camera will form an image which is segmented into foreground and background pixels. Each image is described by pixels, with image j consisting of the set of pixels:

$$I_j = \left\{ i_{0,0}, i_{1,0}, i_{0,1}, ...i_{w_j-1,h_j-1} \right\} \tag{2}$$

Thus a particular pixel in image j will be referred to by $i_\mathbf{p} \in I_j$, or more concisely, $(\mathbf{I}_j)_\mathbf{p}$ where $\mathbf{p} = [p_0 \; p_1]$.

Each camera will effectively project the 3D world into the respective image. Let the transformation performed by camera j be called $T_j(\mathbf{r}) = \mathbf{p}$, where $\mathbf{r}$ is a 3D point in the real world and $\mathbf{p}$ is the 2D integer vector which is used to index the pixels in the image.

Finally let there be a function U such that $U_j(\mathbf{p}, z) = \mathbf{r}$ which, for a given pixel index $\mathbf{p}$ in image j, gives the 3D point $\mathbf{r}$ at an orthogonal distance of z from the camera.

In 2.75D the resulting data structure is defined as:

$$M = \{M_1, M_2, ...M_n\} \tag{3}$$

$$M_j = \{m_{0,0}, m_{1,0}, m_{0,1}, ...m_{w_j-1,h_j-1}\} \tag{4}$$

Here each element m in M_j is used to represent how each pixel in I_j is reconstructed. This is achieved by allowing each element m to be the set of all orthogonal distances that could be in the original subject. The distances are the actual real values ($\in \mathbb{R}$), thus there is no additional loss of information due to discretisation.

The algorithm can thus be demonstrated by evaluating the set of depths of each pixel p in image j:

$$z \in (\mathbf{M}_j)_\mathbf{p} \quad \text{iff} \quad (I_k)_\mathbf{q} = 1 \quad \forall\, k \text{ st} \begin{cases} 0 \le q_0 < w_k \\ 0 \le q_1 < h_k \end{cases} \\ \text{where } \mathbf{q} = T_k(U_j(\mathbf{p}, z)) \tag{5}$$

For simplicity, we do not test for the full cross-section of the projected ray being completely within the subject pixels in all of the other views, but just evaluates the central point of the cross-section. The full test can be achieved by testing the pixels which contribute to the quadrilateral formed by the projection of the four corners of the source pixel.

It is wished to project the ray at an optimal rate so that correspondence checks between views are neither duplicated or missed. The ray can be modelled as a line:

$$\begin{bmatrix} x_{3d} \\ y_{3d} \\ z_{3d} \end{bmatrix} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} + \lambda \begin{bmatrix} d \\ e \\ f \end{bmatrix} \tag{6}$$

where $[a \;\; b \;\; c]^T$ is the source camera's origin, and $[(a+d) \quad (b+e) \quad (c+f)]^T$ is a point on the ray whose z value is 1 unit from that source camera, both with respect to the other camera. A 3D point on the line is mapped onto the destination image (the epipolar line) by:

$$\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{z_{3d}} \begin{bmatrix} f_x x_{3d} \\ f_y y_{3d} \end{bmatrix} = \begin{bmatrix} f_x \frac{a+\lambda d}{c+\lambda f} \\ f_y \frac{b+\lambda e}{c+\lambda f} \end{bmatrix} \tag{7}$$

where f_x and f_y are the effective focal lengths in the x and y directions. Thus as λ increases, various pixels on the destination view are visited, described by the respective discretised epipolar line. However, it is required that all pixels are visited the minimum number of times. By analysing these equations the amount by which λ must increase, $\delta\lambda$, can be formulated. The equation for the required change in λ along the image's x-axis is:

$$\delta\lambda = \left| \begin{array}{l} \frac{(c+\lambda f)^2}{f_x|(af-cd)|-f(c+\lambda f)} \\ \infty \end{array} \right. \text{for} \begin{array}{l} cd \neq af \\ cd = af \end{array} \tag{8}$$

A similar equation for the image's y-axis can also be formulated, and for a given λ it is the minimum of the two that should be selected. These equations thus yield the optimum rate by which λ must be increased. For many views, it is the minimum change in λ over all of the possible destination views that is selected.

Figure 1 demonstrates the results of VI using the 3D voxel and 2.75D methods. The results from this new representation are similar to the 3D voxel-based solutions, except that they are analysed at the most suitable level of resolution. The error visible is due to the approximation by projecting a line and not a pyramid.

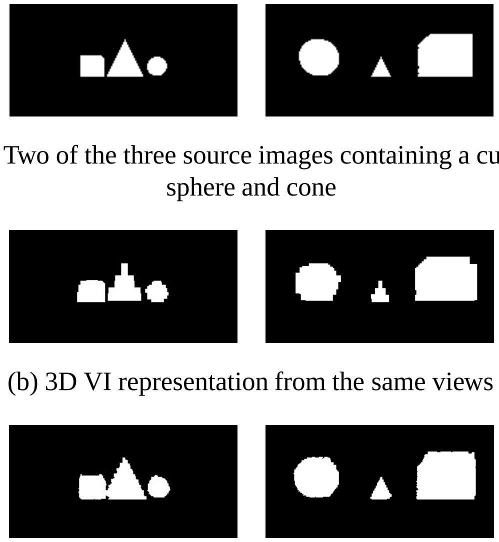
4.2 Colour and grey-scale implementation

A similar method to introduce colour and grey-scale as the voxel-based algorithm is used, but a weighting is now used instead of voxel sides, based upon the dot-produce between the necessary rays:

$$2w - 1 = \cos\theta = \frac{\mathbf{r_s} \cdot \mathbf{r_d}}{|\mathbf{r_s}||\mathbf{r_d}|} \tag{9}$$

where $\mathbf{r_s}$ is the vector from the source view to the 3D point and $\mathbf{r_d}$ is the vector from a destination view to the 3D point. This weighting affects the measure of confidence, thus:

$$m = \frac{\sigma^2 + k_1}{n + k_2} \tag{10}$$



(a) Two of the three source images containing a cube, sphere and cone

(b) 3D VI representation from the same views

(c) Results of the new 2.75D representation

Figure 1. Comparison between the voxel-based and new representations.

$$\sigma^2 = \frac{1}{n}\sum_{i=1}^{v}(w_i p_i^2) - \left(\frac{1}{n}\sum_{i=1}^{v}(w_i p_i)\right)^2 \tag{11}$$

$$n = \sum_{i=1}^{v} w_i \tag{12}$$

where v is the number of views, w_i is the weighting of the pixel from view i, and p_i is that pixel's shade. It is trivial to incorporate colour into the measure: As with the voxel-based system, the grey-scale and colour algorithms are iterative. To remove the background, the segmented images obtained from the 2D background removal are used to mask the 2.75D data.

5. Extraction

We use template matching, the dual of the Hough Transform, to extract moving objects, making use of Genetic Algorithms to enable the high dimensional parameter spaces to be searched. Such a method was used by Cunado *et al* [1] for 2D human gait analysis. The templates are defined using constructive solid graphics (CSG), and are dynamic with respect to time.

6. Example

To analyse gait, the harmonics in the upper legs are extracted, modelled by two cylinders that oscillate about the hip, requiring a total of 23 parameters. The 2.75D and 3D algorithms both extracted a gait pattern, with the former being more accurate. The 2D algorithm failed due to poor background removal which is the key to its success.

Figure 2 shows a selection of the source images and the processed scenes. Figures 2d,g demonstrate that a voxel 3D filter does not perform as well as the 2D filter, and thus an improvement would be to filter out everything that does not lie within the hull of the object. Again, comparing figure 2c with figure 2f, the clarity of the new 2.75D method presents itself. The main source of noise in these images is parts of the background which are of the same colour as the trousers.

7. Conclusions and further work

Three systems have been described which are able to extract and describe 3D dynamic objects in abstract real-world scenes. However the 2D system has been shown not to be applicable when the objects cannot be segmented or are significantly complicated, for example self-occluding gait, but it does perform well for poorly calibrated cameras. The 3D system is much faster, but the reduced data is costly on the parameterisation, once again, especially for complicated models. The 2.75D system, based upon a new data representation, has successfully extracted more complicated models, using colour to obtain *better-than-hull* data. Unfortunately it was slower by a factor of five for our three camera set-up compared with the 3D system, but the extraction was approximately of the same order as the 2D system, although noting that it does have the overhead of scene construction.

References

[1] D. Cunado, J. Nash, M. Nixon, and J. Carter. Gait extraction and description by evidence–gathering. *Proceedings of AVBPA 99*, pages 43–48, 1999.

[2] A. Laurentini. The visual hull concept for silhouette–based image understanding. *IEEE Transactions on PAMI*, **16**(2):150–162, 1994.

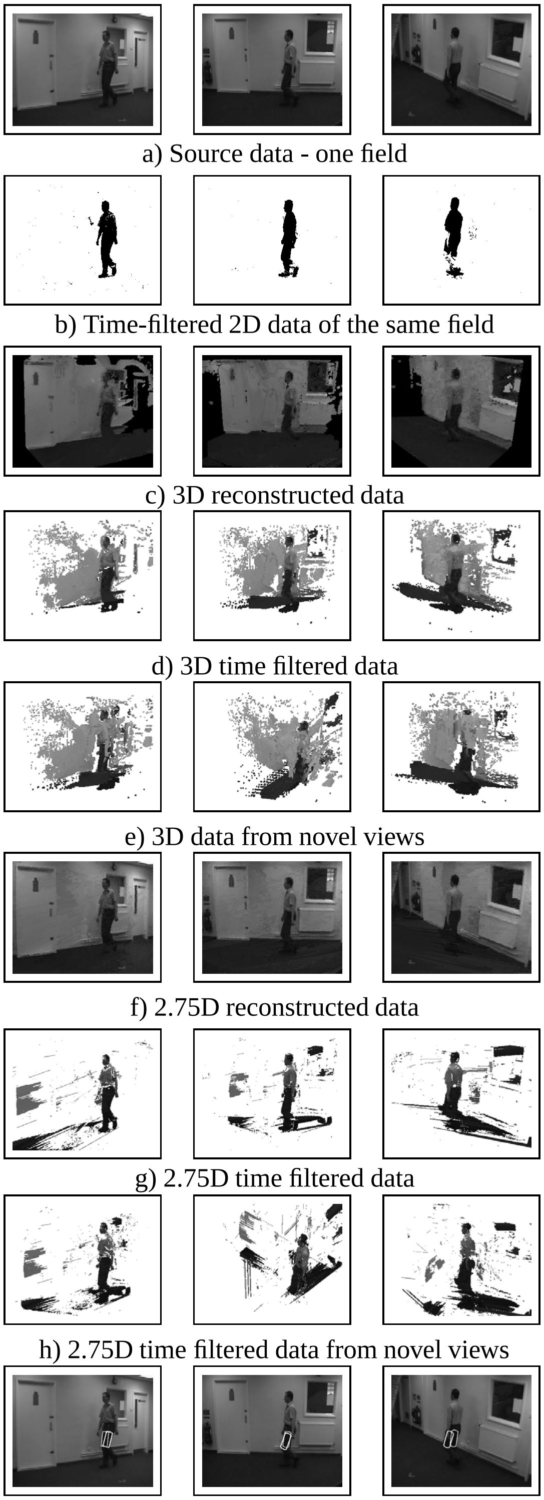
[3] W. Martin and J. Aggarwal. Volumetric descriptions of objects from multiple views. *IEEE Transactions on PAMI*, **5**(2):150–158, 1983.

[4] W. Matusik, C. Buehler, R. Raskar, S. Gortler, and L. McMillan. Image-based visual hulls. In *Siggraph 2000, Computer Graphics Proceedings*, pages 369–374, 2000.

[5] J. Nash, J. Carter, and M. Nixon. Dynamic feature extraction via the velocity Hough transform. *Pattern Recognition Letters*, **18**:1035–1047, 1997.

[6] S. Seitz and C. Dyer. Photorealistic scene reconstruction by voxel coloring. *Internal Journal of Computer Vision*, **35**(2):151–73, 1999.

We gratefully acknowledge partial support by the European Research Office of the US Army under Contract No. N68171-01-C-9002.



a) Source data - one field

b) Time-filtered 2D data of the same field

c) 3D reconstructed data

d) 3D time filtered data

e) 3D data from novel views

f) 2.75D reconstructed data

g) 2.75D time filtered data

h) 2.75D time filtered data from novel views

i) 2.75D result overlaid onto originals

Figure 2. The human gait extraction.