

3-D Shape Recovery Using A Deformable Model

Xinquan Shen and David Hogg
School of Computer Studies
The University of Leeds
Leeds, LS2 9JT, UK
{shen,dch}@uk.ac.leeds.scs

Abstract

The paper describes a method for recovering the 3-D shape of a moving object from a sequence of images. While following the motion of the object, a 3-D surface model, initialised to be spherical, progressively deforms itself under the action of simulated external forces applied to the model to drive its profile towards the object profile extracted from the image. An internal energy coupled to the model encourages a smooth and uniform deformation. The model is also constrained to be symmetrical about a plane parallel to the direction of motion. Poses for the model to correctly align with the object in 3-D are computed by using the motion trajectory of the object in the image plane and a non-linear least squares method. Experimental results are presented for recovery of shape models of vehicles. The resulting model represents the 3-D object shape, and may be used for several purposes including object recognition, tracking and visualisation.

1 Introduction

In previous work [1], we proposed a deformable model-based method for recovering object shape from a sequence of images, assuming the object is rigid, mirror symmetric, and moves on a ground plane. The method uses a physically based model similar to that proposed by Terzopoulos, *et al* [2, 3] coupled with special internal constraints. Shape recovery is achieved through deformation of the model in response to image profiles, by its alignment with the object in 3-D using motion trajectory information. This paper extends the previous work in two ways: (1) an internal energy is incorporated to encourage smooth and uniform deformation of the model, constraining parts of the model surface which are not directly influenced by external forces, (2) a more robust method is used for posing the model to align with the object in 3-D.

The method uses a 3-D physically based surface model whose behaviour is determined by its internal energy and the external forces acting on it. By initially taking the object shape as a sphere, the model follows the motion of the object and actively deforms itself in response to the action of simulated external forces computed from each image in the sequence, to approach the object shape. For

each image, the model deforms away from its current state which is the best approximation to the object shape up to the previous image. The simulated external forces are applied to the model in such a way that they drive the model profile towards the object profile extracted from the image. The coupled internal spline energy encourages a smooth and uniform deformation. External forces are imposed symmetrically on the model to maintain mirror symmetry about a plane parallel to the direction of motion of the object. Alignment of the model with the object in 3-D is achieved by estimating the pose of the object for each image using back-projection of its image trajectory and a non-linear least squares method.

2 Related Work

The 3-D locations of points may be inferred from sequential locations of feature points (e.g. corners) in a 2-D sequence of images. The object surface is then approximated by either triangulation of these points [4, 5] or using a patchwork of parametric surface functions. Here the detection of the feature points and the establishment of their correspondence which are sensitive to noise perturbation are essential. Producing a detailed and smooth representation of surface shape is nontrivial. The extension to deal with closed surfaces is also not straightforward.

Methods for producing 3-D object shape using 2-D object profiles extracted from multiple images have been proposed, for example, by Chien and Aggarwal [6], and recently by Stenstrom and Connolly [7]. The methods are based on the intersections of 3-D bounding volumes produced by sweeping 2-D object profiles (regions) along the line of sight. These are applied in a controlled environment so that images are obtained from predetermined known views.

Recently, physically based models have been proposed for modelling rigid and non-rigid objects [8, 9, 10]. By combining the dynamics, the model can accommodate both object shape and its motion. Shape recovery from a given range data image is achieved by applying suitable external forces to the model [3, 11]. Our method uses a similar physically based model, incorporating suitable internal forces. Shape recovery is achieved by aligning the model with the object in 3-D and imposing external forces derived from the object profiles.

3 Deformable Model

3.1 Coordinate systems

A world coordinate system \mathbf{X} is chosen so that the ground plane on which objects are moving is given by $Z = 0$ (Fig. 1a). Scene points are expressed in homogeneous form so that $(\lambda X, \lambda Y, \lambda Z, \lambda)$, $\lambda \neq 0$ represents the point (X, Y, Z) . The transformation from world coordinates to pixel coordinates \mathbf{x} of the image, again expressed in homogeneous form, is characterised by a 3×4 calibration matrix \mathbf{C}

$$\mathbf{x} = \mathbf{C}\mathbf{X} \quad (1)$$

\mathbf{C} is computed by examining the distribution of projected object height in the images [12]. Finally, the model is expressed in a model centered coordinates system \mathbf{X}' .

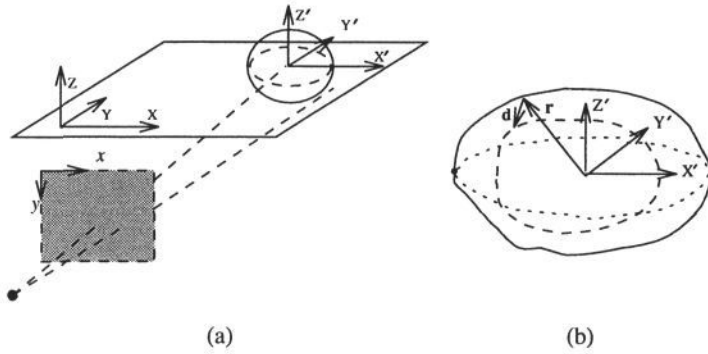


Figure 1: (a) Definitions of coordinate systems; (b) Deformable model

3.2 Deformable model

The physically based model, $\mathbf{r}(u, v)$, is a closed surface (topologically to an equivalent sphere), where, $(u, v) \in [0, 1]^2$ are the material coordinates. It has boundary conditions which constrain the model to be “seamed” at the curves $v = 0$ and $v = 1$, and with two poles [1]. For each image, the model deforms away from its current state, which is the best approximation to the object shape up to the previous image, to balance the action of the external forces. The deformation of the model is characterised by a displacement $\mathbf{d}(u, v)$ away from its previous position (Fig. 1b).

Suppose the model is not subject to any shift during the deformation. Under the static situation, its deformation behaviour is governed by the following equation [1]:

$$\delta\mathcal{E} = \mathbf{f} \quad (2)$$

where, \mathbf{f} is the external force acting on the model; \mathcal{E} is the deformation energy associated with the model. The variational derivative $\delta\mathcal{E}$ is thus the elastic force so produced.

For our purpose, the elastic force $\delta\mathcal{E}$ is taken as following

$$\delta\mathcal{E} = \delta_{\mathbf{d}}\mathcal{E}_1(\mathbf{d}) + \delta_{\mathbf{r}}\mathcal{E}_2(\mathbf{r}) \quad (3)$$

where, $\mathcal{E}_1(\mathbf{d})$ is a deformation energy produced due to the displacement. $\mathcal{E}_2(\mathbf{r})$ is an internal energy from the current state of the model to maintain the intrinsic property of the model (e.g. coherence and smoothness of the surface). Variational derivatives $\delta_{\mathbf{d}}\mathcal{E}_1(\mathbf{d})$ and $\delta_{\mathbf{r}}\mathcal{E}_2(\mathbf{r})$ are thus the elastic forces so produced.

The deformation energy $\mathcal{E}_1(\mathbf{d})$ used here is the membrane deformation energy suggested by Terzopoulos [3].

$$\mathcal{E}_1(\mathbf{d}) = \frac{1}{2} \int \left(w_1 \left(\left(\frac{\partial \mathbf{d}}{\partial u} \right)^2 + \left(\frac{\partial \mathbf{d}}{\partial v} \right)^2 \right) + w_0 \mathbf{d}^2 \right) dudv \quad (4)$$

Constants w_0 and w_1 control the local magnitude and variation of the deformation respectively. This deformation energy ensures that individual deformations at each step are smooth but may accumulate over several frames to allow creases to emerge.

To prevent the unwanted creases, especially on the local surfaces never projectly close to profile, a spline energy is taken as the internal energy $\mathcal{E}_2(\mathbf{r})$ to encourage a smooth and uniform model surface,

$$\mathcal{E}_2(\mathbf{r}) = \frac{1}{2} \int c \left(\left(\frac{\partial^2 \mathbf{r}}{\partial u^2} \right)^2 + \left(\frac{\partial^2 \mathbf{r}}{\partial v^2} \right)^2 \right) dudv, \quad (5)$$

where, constant c is a weighting parameter.

The variational derivatives of these energy are then [13]:

$$\begin{aligned} \delta_{\mathbf{d}} \mathcal{E}_1(\mathbf{d}) &= w_0 \mathbf{d}(u, v) - w_1 \left(\frac{\partial^2 \mathbf{d}(u, v)}{\partial u^2} + \frac{\partial^2 \mathbf{d}(u, v)}{\partial v^2} \right) \\ \delta_{\mathbf{r}} \mathcal{E}_2(\mathbf{r}) &= c \left(\frac{\partial^4 \mathbf{r}(u, v)}{\partial u^4} + \frac{\partial^4 \mathbf{r}(u, v)}{\partial v^4} \right) \end{aligned} \quad (6)$$

For each image, the external forces are computed and applied to the points on the model which project onto the model profile, and therefore drive the model profile towards the object profile. Other points on the model are adjusted by virtue of the internal energy and the propagation of the external forces. The external force on a point \mathbf{X} on the model is computed based on the Euclidean distance from its image \mathbf{x} to object profile (see [14] for details about the computation of the external force and the extraction of the object profile). In the numerical implementation of this method, the external forces are computed at each iteration step. To maintain the mirror symmetry constraint, external forces are always applied to symmetry points also.

3.3 Numeric Implementation

The system (2) is discretised for numeric simulation. By discretising the domain $0 \leq u, v \leq 1$ into a regular $M \times N$ mesh of nodes and using finite differences to approximate the derivatives, we have the following discrete approximation of the governing equation:

$$\mathbf{K}_0 \mathbf{D} + \mathbf{K}_1 \mathbf{R} = \mathbf{F} \quad (7)$$

where, \mathbf{K}_0 and \mathbf{K}_1 are stiffness matrices

$$\mathbf{K}_0 = \begin{bmatrix} \mathbf{K}_{0x} & 0 & 0 \\ 0 & \mathbf{K}_{0y} & 0 \\ 0 & 0 & \mathbf{K}_{0z} \end{bmatrix}, \mathbf{K}_1 = \begin{bmatrix} \mathbf{K}_{1x} & 0 & 0 \\ 0 & \mathbf{K}_{1y} & 0 \\ 0 & 0 & \mathbf{K}_{1z} \end{bmatrix}$$

$\mathbf{K}_{ix} = \mathbf{K}_{iy} = \mathbf{K}_{iz}, i = 0, 1$ are $MN \times MN$ matrices. \mathbf{R} , \mathbf{D} and \mathbf{F} are the vectors of model nodes, displacements at model nodes and external forces acting on the model nodes respectively,

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_x \\ \mathbf{R}_y \\ \mathbf{R}_z \end{bmatrix}, \mathbf{D} = \begin{bmatrix} \mathbf{D}_x \\ \mathbf{D}_y \\ \mathbf{D}_z \end{bmatrix}, \mathbf{F} = \begin{bmatrix} \mathbf{F}_x \\ \mathbf{F}_y \\ \mathbf{F}_z \end{bmatrix}$$

Each element in these vectors is an $M \times N$ vector. \mathbf{K}_0 and \mathbf{K}_1 are symmetrical, sparse and non-singular, and are only computed once. The iteration equation for updating the model is:

$$\mathbf{R}^{n+1} = (\mathbf{I} - \mathbf{K}_0^{-1} \mathbf{K}_1) \mathbf{R}^n + \mathbf{K}_0^{-1} \mathbf{F}^n \quad (8)$$

Our method for shape recovery requires the correct alignment of the model with the object in 3-D before performing the deformation operations at each image. This can be achieved by finding poses for the model which are consistent (up to a relative depth) with the object poses in 3-D. Since the rigid object is moving on the ground plane, these poses are expressed in terms of orientation θ about the Z' -axis and positions (P_x, P_y, P_z) (P_z is fixed) of the model coordinate system within the world coordinate system, and are represented by a rigid transformation.

Since the model is progressively deformed to be consistent with the object profile, it seems sensible to find the pose parameters which transform the model to a state which has a minimum distance measurement between its profile and object profile. However, experiments found this can only find the approx position parameters, since there is little orientation information directly available from the individual object profile. Instead, we compute the orientation according to the object trajectory in the image plane, and then estimate the position parameters by finding a minimum distance measurement between the implied model profile for that orientation and object profile.

4.1 Orientation estimation

The orientation of the model is assumed to align with the direction of motion of the object, and can be approximately computed from its motion trajectory in the image plane approximated by the centroids of regions obtained by picture differencing [15]. Let \mathbf{x}_i and \mathbf{x}_{i+1} be the centroids of object regions in images i and $i+1$ respectively, and assume they are the images of two points with the same height to the ground, h , i.e.

$$\begin{cases} \mathbf{x}_i &= \mathbf{C} [X_i \ Y_i \ h \ 1]^t \\ \mathbf{x}_{i+1} &= \mathbf{C} [X_{i+1} \ Y_{i+1} \ h \ 1]^t \end{cases} \quad (9)$$

X_i, X_{i+1}, Y_i and Y_{i+1} can be solved from the above equation. The orientation of the model for image $i+1$ is then determined by

$$\theta = \arctan2(X_{i+1} - X_i, Y_{i+1} - Y_i) \quad (10)$$

This can be a good estimation when the distance between the points (X_i, Y_i, h) and (X_{i+1}, Y_{i+1}, h) is not too small. More robustly, we estimate the orientation by computing the tangent at the point (X_{i+1}, Y_{i+1}, h) , which can be estimated by interpolating the points (X_i, Y_i, h) , (X_{i+1}, Y_{i+1}, h) and (X_{i+2}, Y_{i+2}, h) with a 2nd order parametric curve.

4.2 Position estimation

In our previous work, the position of the model is determined by taking the centroid of the object region in the image as the projection of the model center. However, this simple method cannot effectively cope with situations where the extracted object regions are not compact. Here, we use a more robust method to estimate the position parameters.

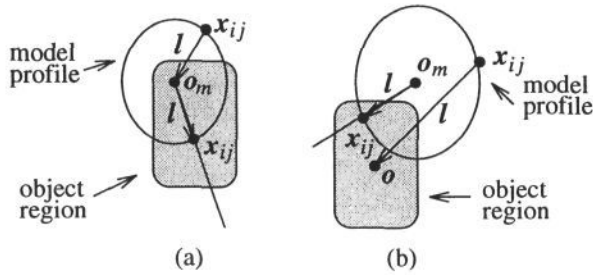


Figure 2: Determination of direction l . (a) o_m is inside object region; (b) o_m is outside object region.

In the deformation operations, the model is deformed until its profile is consistent with the object profile up to certain criterion. This, at least, makes the part of model surface be the approximation of the object surface. Since the difference of visible surfaces of the object between two frames is relatively small, we estimate position parameters by using model itself.

Let $N_i, i = 1, 2, \dots, N$ be the profile of the model obtained to date; $\mathbf{X}_{ij}(P_x, P_y), j = 1, 2, \dots, L$ be the L nodes on the model which originally all project to the same image point N_i . Define a merit function $\Psi(P_x, P_y)$

$$\Psi(P_x, P_y) = \sum_{i=1}^N \phi_i^2 \quad (11)$$

where, ϕ_i depends on (P_x, P_y) and $\phi_i = \min_j \{\delta(\mathbf{X}_{ij}(P_x, P_y))\}$. $\delta(\mathbf{X}_{ij}(P_x, P_y))$ is the distance from the x_{ij} , the image of $\mathbf{X}_{ij}(P_x, P_y)$, to the boundary of object region in the current image along a direction l . Direction l is determined according to whether the x_{ij} and the image of model center, o_m , are inside the object region (see Fig. 2). If the o_m is inside the object region, the l is along the vector from o_m to x_{ij} , or from x_{ij} to o_m depending whether x_{ij} is inside the object region; If the o_m is outside the object region, l is along the vector from o_m to x_{ij} when x_{ij} is inside the object region, or along the vector from x_{ij} to the center of object region, o , when x_{ij} is outside the object region.

Write ϕ_i^2 as

$$\phi_i^2 = (x_i - x'_i)^2 + (y_i - y'_i)^2 \quad (12)$$

(x_i, y_i) is an image point of one of the \mathbf{X}_{ij} , (x'_i, y'_i) is a found point on the boundary of object region. The merit function (11) then can be rewritten as

$$\Psi(P_x, P_y) = \sum_{i=1}^{2N} \varphi_i^2(P_x, P_y) \quad (13)$$

where,

$$\varphi_i(P_x, P_y) = \begin{cases} x_i - x'_i, & i = 2m - 1, \quad m = 1, 2, \dots, N \\ y_i - y'_i, & i = 2m, \quad m = 1, 2, \dots, N \end{cases} \quad (14)$$

Now our aim is to find parameters (P_x, P_y) (with fixed P_z and θ) which minimise the merit function (13).

Transformation from 3D to 2D is a non-linear operation, but, it is a smooth and well-behaved transformation to allow the use of Newton least squares method to estimate unknown parameters P_x and P_y [16, 17]. In the implementation, the Levenberg-Marquardt method is used [18]. Experiments found the estimation normally converges to the correct solution if the method begins with an initial guess which is not far away from the true value (in our experiment, the initial guess is the position for the previous image).

4.3 Determination of start pose

To begin shape recovery, a start pose for the initial model which is a sphere needs to be suitably determined. With the first two images in the sequence, we determine a pose of the model at the second image, and begin the operations of shape recovery with this image.

The position parameters can be determined by setting the model to a position where the projection of the model center, \mathbf{X}_0 , overlaps with the centroid of the object region in the image, \mathbf{x}_0 , i.e.

$$\mathbf{x}_0 = C\mathbf{X}_0 \quad (15)$$

By presetting the height of the model center to the ground, h , the 3-D position of the model center can be determined. Since the pose for the model is determined up to a relative depth to conform with object pose in 3-D, the selection of the height h could be arbitrary. The orientation for the model at the second image can be determined with the method in section 4.1.

The object to be modelled is assumed to be symmetrical about a plane which is vertical to the ground plane and parallel to the direction of motion. For the convenience of imposing the external forces symmetrically, the initial model is then posed so that it has the start pose, and all the points with material coordinate $v = 0$ are on the symmetry plane.

5 Experimental Results

The above method is applied to 3 sequences of images depicting 3 different cars turning into a parking space. The model is discretised into a 17×32 mesh of nodes, and the height of its center to the discussed ground plane is set to zero. All the results are obtained using the same set of parameter values.

Figure 3a shows 4 images from a sequence of 24 images (256×192 pixels) of a Volkswagen, and figure 3b shows the initial model and the intermediate state of the model after processing of the same images. The last model is the final shape model for the object. Figure 4 shows 4 images from a sequence of 25 images (180×143 pixels) of a Fiat Uno, and the corresponding intermediate models. The model at frame 25 is the final model. Figure 5 show images of another car and its shape model.

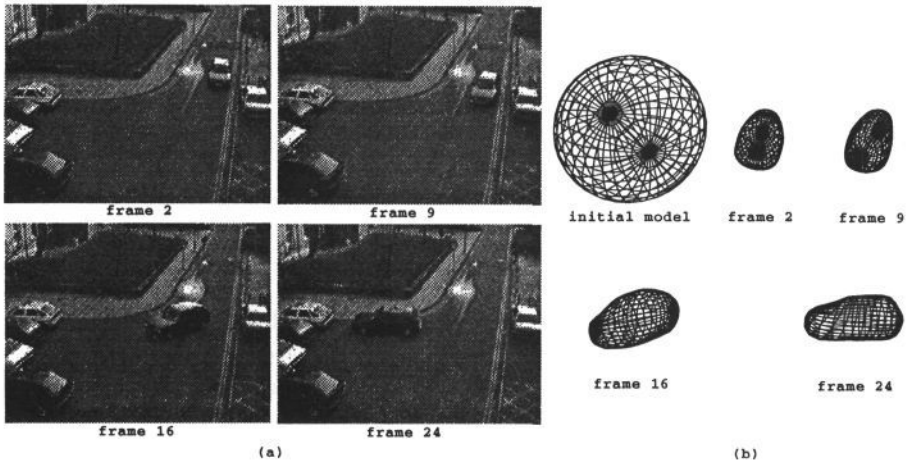


Figure 3: (a) Four object (Volkswagen) images from a sequence of 24 images. (b) Initial model and intermediate models corresponding to the images in (a); the model at frame 24 is the final model

6 Conclusion

We have proposed a method for generating 3-D object shape models from 2-D images using a deformable model. While following the motion of the object, the model progressively deforms itself under the action of external forces derived from the image, and approaches the object shape. An internal energy is incorporated in the model to encourage smooth and uniform deformation and to constrain these parts of surface not directly influenced by the external forces. Alignment of the model with the object in each image is achieved using the motion trajectory of the object in the image plane and a non-linear least squares method. The result is a 3-D model suitable for use in a variety of applications, such as object recognition, tracking and virtual reality.

Acknowledgement

X. Shen gratefully acknowledges the financial support from the Chinese government and the British Council.

References

- [1] X. Shen and D. Hogg. Shape Models from Image Sequences. In Jan-Olof Eklundh, editor, *Third European Conference on Computer Vision*, volume I, pages 225–230, 1994.
- [2] D. Terzopoulos, A. Witkin, and M. Kass. Constraints on Deformable Models: Recovering 3D Shape and Nonrigid Motion. *Artificial Intelligence*, 36(1):91–123, 1988.

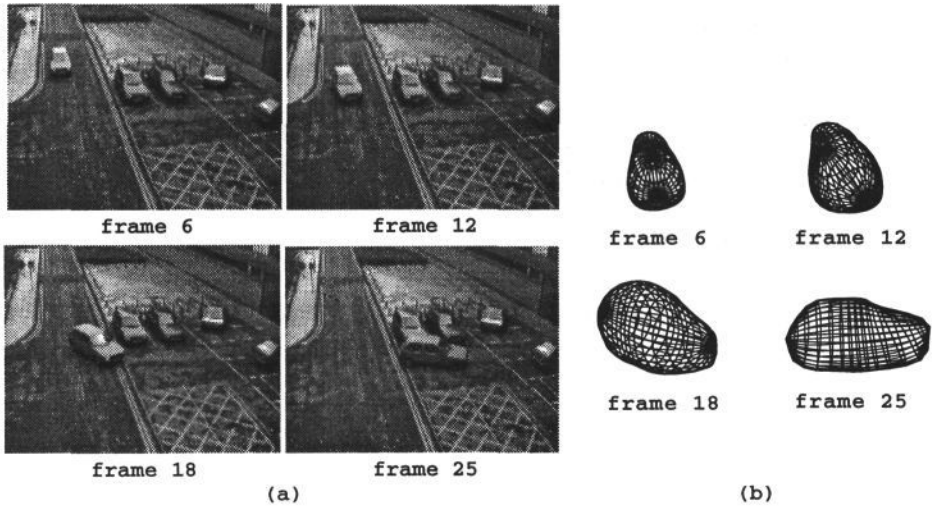


Figure 4: (a) Four object (Fiat Uno) images from a sequence of 25 images. (b) Intermediate models corresponding to the images in (a); the model at frame 25 is the final model

- [3] D. Terzopoulos and D. Metaxas. Dynamic 3D Model with Local and Global Deformations: Deformable Superquadrics. In *Proceedings of the Third International Conference on Computer Vision (ICCV 90)*, pages 606–615, Osaka, Japan, 1990.
- [4] D. Charnley and R. Blissett. Surface Reconstruction from outdoor Image Sequences. *Image and Vision Computing Journal*, 7(1):10–16, 1989.
- [5] H. Westphal and H.-H. Nagel. Toward the Derivation of Three-dimensional Descriptions from Image Sequences for Nonconvex Moving Objects. *Computer Vision, Graphics and Image Processing*, 34:302–320, 1986.
- [6] C. H. Chien and J. K. Aggarwal. A Volume/Surface Octree Representation. In *Proceedings of 7th International Conference on Pattern Recognition*, pages 817–820, Montreal, August 1984.
- [7] J. R. Stenstrom and C. I. Connolly. Construction Object Models from Multiple Images. *International Journal of Computer Vision*, 9(3):185–212, 1992.
- [8] D. Terzopoulos, J. Platt, A. Barr, and K. Fleischer. Elastically Deformable Models. *ACM Computer Graphics*, 21(14):205–214, 1987.
- [9] D. Terzopoulos and A. Witkin. Physically-based Models with Rigid and Deformable Components. *IEEE Computer Graphics and Applications*, 8(6):41–51, 1988.
- [10] A. P. Pentland. Perceptual Organization and the Representation of Natural Form. *Artificial Intelligence*, 28:193–331, 1986.

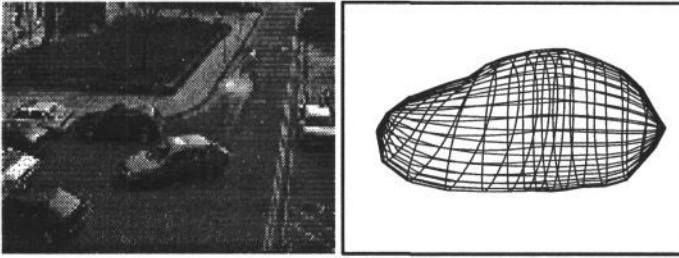


Figure 5: An object image in a sequence and the shape model of the object

- [11] A. Pentland and S. Sclaroff. Closed-form Solution for Physically Based Shape Modelling and Recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-13(7):715–729, 1991.
- [12] D. Hogg, D. Young, and L-Q Xu. Statistical Regularity in Motion Sequences. Technical report, School of Computer Studies, The University of Leeds, 1993.
- [13] R. Courant and D. Hilbert. *Methods of Mathematical Physics II*. Interscience, London, 1953.
- [14] X. Shen and D. Hogg. Shape models from image sequences. Technical Report Report 93.37, School of Computer Studies, University of Leeds, Leeds, LS2 9JT, UK, October 1993.
- [15] A. Baumberg and D. Hogg. Learning Flexible Models from Image Sequences. In Jan-Olof Eklundh, editor, *Third European Conference on Computer Vision*, volume I, pages 299–308. Springer-Verlag, 1994.
- [16] D. G. Lowe. Fitting Parameterized Three-Dimensional Models to Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-13(5):441–450, May 1991.
- [17] J-L Chen, G. C. Stockman, and K. Rao. Recovering and Tracking Pose of Curved 3D Objects from 2D Images. In *CVPR*, pages 233–239, 1993.
- [18] L. E. Scales. *Introduction to Non-linear Optimization*. Macmillan, 1985.