ISSN1000-9825 Journal of Software 软件学报 2000,11(4):435~440

A Model-Based Approach to Human Animation*

LAO Zhi-qiang PAN Yun-he

(State Key Laboratory of CAD&CG Zhejiang University Hangzhou 310027)
(Institute of Artificial Intelligence Zhejiang University Hangzhou 310027)
E-mail: aszqlao@ntu.edu.sg

Abstract In this paper, starting from point-point and line-line correspondence, a detailed description of the calculation of projection parameters for 2D video features and 3D model features correspondence is presented firstly. Then a detailed description of the calculation of projection parameters for 2D human video features and 3D human model features correspondence is given. At the same time, an animation example of human walking is introduced which is produced by an animation system called Video&Animation Studio. The system is designed according to the above idea.

Key words Human animation, correspondence between video features and model features.

As we know, there are two categories for the current computer vision systems: Depth Reconstruction based Systems (DRS)[1~4] and Knowledge based Systems (KS)[5~7]. DRS extract image features (such as spatial vision, motion, texture, etc.) from the image, then form a 2.5D sketch and match these features with 3D model. In DRS, all the recognition work was done in 3D space while in KS perceptional organization was employed to form 2D Perceptional Structure Groupings from image features, then knowledge match procedure is used to get mapping parameters with 3D model.

We try to find a method for bridging the gap between 2D images and knowledge of 3D objects without any preliminary derivation of depth. A quantitative method is used to simultaneously determine the best viewpoint and object parameter values for fitting the projection of a 3D model to given 2D features. This method allows a few initial hypothesized matches to be extended by making accurate quantitative predictions for the locations of those object features in the image. This provides a highly reliable method for verifying the presence of a particular object, since it can make use of the spatial information in the image to the full degree of available resolution. The final judgment as to the presence of the object can be based on only a subset of the predicted features, since the problem is usually greatly over-constrained due to the large number of visual predictions from the model compared with the number of free parameters.

1 Solving for Spatial Correspondence

Many areas of AI are aimed at the interpretation of data by finding consistent correspondences between the

* This research is supported by the National Natural Science Foundation of China (国家自然科学基金, No. 69233011) and by the National High Technology Development Program of China (国家 863 高科技项目基金, No. 863-306-04-03-3). **LAO Zhi-qiang** was born in 1967. He is now a research fellow of Centre for Graphics and Imaging Technology, Nanyang Technological University, Singapore. His current research areas include computer animation, computer vision, artificial intelligence, imagary thinking. **PAN Yun-he** was born in 1945. He is a member of The Chinese Academy of Engineering and the president of Zhejiang University. His current research areas include intelligent CAD, computer graphics, image processing, GIS, imagary thinking, artificial intelligence.

Manuscript received 1998-06-22, accepted 1998-11-10.

© 中国科学院软件研究所 http://www.jos.org.cn

data and prior knowledge of the domain. In our application, we need to define the consistency conditions for judging correspondence between image data and 3D knowledge. Unlike many other areas of AI, an important component of this knowledge is quantitative spatial information that requires specific mathematical techniques for achieving correspondence. The particular constraint that we wish to apply can be stated as follows:

The locations of all projected model features in an image must be consistent with the projection from a single viewpoint.

The precise problem we wish to solve is as follows: given a set of known correspondences between 3D model points and 2D image points, what are the values of the unknown projection and model parameters that will result in the projection of the given model points into the corresponding image points.

The approach taken in this paper is to linearize the projection equations and apply Newton's method for the necessary number of iterations.

1.1 Application of Newton's method

In the field of CG we can describe the projection of a 3D model point $p(x,y,z)$ into a 2D image point (u, v) with the following equations:

$$(x,y,z)=R(p-t), \tag{1}$$

$$(u,v)=\left(\frac{fx}{z},\frac{fy}{z}\right), \tag{2}$$

where t is a 3D translation vector, R is the rotation matrix which transforms point p in the original model coordinates into a point (x,y,z) in camera-centered coordinates. These are combined in the second equation with a parameter f proportional to the camera focal length to perform perspective projection into an image point (u,v).

Our task is to solve for t, R and possibly f, given a number of model points and their corresponding locations in an image. In order to apply Newton's method, we must be able to calculate the partial derivatives of u and v with respect to each of the unknown parameters. However, it is difficult to calculate these partial derivatives for this standard form of the projection equation.

The partial derivatives with respect to the translation parameters can be most easily calculated by first reparameterizing the projection equations to express the translations in terms of the camera coordinate system rather than model coordinates. This can be described by the following equations:

$$(x,y,z)=Rp, \tag{3}$$

$$(u,v)=\left(\frac{fx}{z+D_z}+D_x,\frac{fy}{z+D_z}+D_y\right). \tag{4}$$

Here the variables R and f remain the same as in Eqs. (1) and (2), but the vector t has been replaced by the parameters, D_x, D_y and D_z. The two transforms are equivalent when

$$t=R^{-1}\left[\frac{-D_x(z+D_z)}{f},\frac{-D_y(z+D_z)}{f},-D_z\right]^T. \tag{5}$$

In the new parameterization, D_x and D_y simply specify the location of the object on the image plane and D_z specifies the distance of the object from the camera. As will be shown below, this formulation makes the calculation of partial derivatives with respect to the translation parameters almost trivial.

We are still left with the problem of representing the rotation R in terms of its three underlying parameters. The solution to this second problem is based on the realization that Newton's method does not in fact require an explicit representation of the individual parameters. Therefore, we have chosen to take the initial specification of R as given and add to it incremental rotations φ_x, φ_y, and φ_z about the x, y, z axes of the current camera coordinate system.

Another advantage of using the φs as the convergence parameters is that the derivatives of x, y, z (and therefore of u and v) with respect to them can be expressed in a strikingly simple form. For example, the

© 中国科学院软件研究所 http://www.jos.org.cn

derivative of x at a point with respect to a counterclockwise rotation of φ_z about the z axis is simply $-y$. This follows from the fact that $(x,y,z)=(r\cos\varphi_z, r\sin\varphi_z, z)$, where r is the distance of the point from the z axis, and therefore $\frac{\partial x}{\partial\varphi_z}=-r\sin\varphi_z=-y$. Table 1 gives these derivatives for all combinations of variables.

Table 1 The partial derivatives of x,y,z with respect to φ_x, φ_y, φ_z

	x	y	z
φ_x	0	$-z$	y
φ_y	z	0	$-x$
φ_z	$-y$	x	0

Using Eqs. (3) and (4), it is now straightforward to accomplish our original objective of calculating the partial derivatives of u and v with respect to each of the original camera parameters. For example, Eq. (4) tells us that (substituting $c=(z+D_z)^{-1}$): $\frac{\partial u}{\partial\varphi_y}=fcx+fc^2x^2=fc(x+cx^2)$, similarly, $\frac{\partial u}{\partial\varphi_z}=-fcy$. All the other derivatives can be calculated in a similar way. Table 2 gives the derivatives of u and v with respect to each of the seven parameters of our camera model.

Table 2 The partial derivatives of u and v with respect to each of the camera viewpoint parameters

	u	v
D_x	1	0
D_y	0	1
D_z	$-fc^2x$	$-fc^2y$
φ_x	$-fc^2xy$	$fc(z+cy^2)$
φ_y	$fc(z+cx^2)$	fc^2xy
φ_z	$-fcy$	fcx
f	cx	cy

Our task in each iteration of the multi-dimensional Newton convergence is be to solve for a vector of corrections: $h=[\triangle D_x,\triangle D_y,\triangle D_z,\triangle\varphi_x,\triangle\varphi_y,\triangle\varphi_z]$. If the focal length f is unknown, then $\triangle f$ would also be added to this vector. Given the partial derivatives of u and v with respect to each variable parameter, the application of Newton's method is straightforward. For each point in the model which should match some corresponding point in the image, we first project the model point onto the image using the current parameter estimates and then measure the error in its position compared with the given image point. The u and v components of the error can be used independently to create separate linearized constraints. Then we have

$$\frac{\partial u}{\partial D_x}\triangle D_x+\frac{\partial u}{\partial D_y}\triangle D_y+\frac{\partial u}{\partial D_z}\triangle D_z+\frac{\partial u}{\partial\varphi_x}\triangle\varphi_x+\frac{\partial u}{\partial\varphi_y}\triangle\varphi_y+\frac{\partial u}{\partial\varphi_z}\triangle\varphi_z=E_u, \tag{6}$$

$$\frac{\partial v}{\partial D_x}\triangle D_x+\frac{\partial v}{\partial D_y}\triangle D_y+\frac{\partial v}{\partial D_z}\triangle D_z+\frac{\partial v}{\partial\varphi_x}\triangle\varphi_x+\frac{\partial v}{\partial\varphi_y}\triangle\varphi_y+\frac{\partial v}{\partial\varphi_z}\triangle\varphi_z=E_v. \tag{7}$$

For each point correspondence we derive two equations. From three point correspondences we can derive six equations and produce a complete linear system which can be solved for all six camera model corrections ($\triangle D_x$, $\triangle D_y$, $\triangle D_z$, $\triangle\varphi_x$, $\triangle\varphi_y$, $\triangle\varphi_z$).

$$D_x=D_x+\triangle D_x,\ D_y=D_y+\triangle D_y,\ D_z=D_z+\triangle D_z,$$
$$\varphi_x=\varphi_x+\triangle\varphi_x,\ \varphi_y=\varphi_y+\triangle\varphi_y,\ \varphi_z=\varphi_z+\triangle\varphi_z. \tag{8}$$

Then use Eq. (8) to shrink by about one step of magnitude of vector $(D_x,D_y,D_z,\varphi_x,\varphi_y,\varphi_z)$, and no more than a few iterations would be needed even for high accuracy.

In most applications of this method we will be given more than three correspondences between model and image. In this case the Gaussian least-squares method can easily be applied.

1.2 Use of line-to-line correspondences

In most applications, the feature correspondence between image and model was given in the form of feature

© 中国科学院软件研究所 http://www.jos.org.cn

lines instead of feature points. In this circumstance, modifications should be made on the above method. We should substitute the error on the right side of Eqs. (6) and (7) with the distance between projection line and image line instead of that between projection point and image point. As the distance between unparallel lines is not a fixed value, so we use the distance from point to line to substitute the distance between lines. We represent line equation as follows: (substituting $k=(\sqrt{m^2+1})^{-1}$)

$$-kmu+kv=d, \tag{9}$$

where d is the perpendicular distance from the origin point to the line, m is the slant of the line, so we have

$$\frac{\partial d}{\partial u}=-km,\ \frac{\partial d}{\partial v}=k. \tag{10}$$

Then we have $\frac{\partial d}{\partial D_x}=-km\frac{\partial u}{\partial D_x}+k\frac{\partial v}{\partial D_x}=-km$.

Similarly we have

$$\begin{aligned}&\frac{\partial d}{\partial D_y}=k,\ \frac{\partial d}{\partial D_z}=fkc^2(xm-y),\ \frac{\partial d}{\partial \varphi_x}=fkc(cxym-cy^2-z),\\&\frac{\partial d}{\partial \varphi_y}=fkc(cxy-mz-cmx^2),\ \frac{\partial d}{\partial \varphi_z}=fkc(x+my),\ \text{where } c=(z+D_z)^{-1}.\end{aligned} \tag{11}$$

Now Eqs. (6) and (7) can be represented as follows

$$\frac{\partial d}{\partial D_x}\triangle D_x+\frac{\partial d}{\partial D_y}\triangle D_y+\frac{\partial d}{\partial D_z}\triangle D_z+\frac{\partial d}{\partial \varphi_x}\triangle \varphi_x+\frac{\partial d}{\partial \varphi_y}\triangle \varphi_y+\frac{\partial d}{\partial \varphi_z}\triangle \varphi_z=E_d. \tag{12}$$

With Eqs. (11) and (12), we can have

$$\begin{aligned}&fkc(cxym-cy^2-z)\triangle\varphi_x+fkc(cxy-mz-cmx^2)\triangle\varphi_y+fkc(x+my)\triangle\varphi_z+\\&-km\triangle D_x+k\triangle D_y+fkc^2(xm-y)\triangle D_z=E_d,\end{aligned} \tag{13}$$

where E_d is the perpendicular distance from the two endpoints of corresponding image line to the projection line. As one line has two endpoints, so there are two equations similar to Eq. (13) for each line correspondence. With three lines, we can have a complete linear system, and the solution is $(\triangle D_x,\triangle D_y,\triangle D_z,\triangle\varphi_x,\triangle\varphi_y,\triangle\varphi_z)$. Then we can adjust vector $(D_x,D_y,D_z,\varphi_x,\varphi_y,\varphi_z)$. With Newton's method employed, the projection parameters in the line-line correspondence condition are reached.

2 The Correspondence between Video Human Features and 3D Model Human Features

2.1 Projection parameters calculation for 3D human model

Firstly, we give the data structure of image point and 3D model point as follows:

Point on image:

```
typedef struct IMAGE_POINT {
                                    float x;
                                    float y;
} IMAGE_POINT;
```

Point on model:

```
typedef struct MODEL_POINT {
                                float x;
                                float y;
                                float z;
} MODEL_POINT;
```

Now we can calculate projection parameters of a certain part in 3D human model. Suppose its corresponding feature lines on the image are

© 中国科学院软件研究所 http://www.jos.org.cn

$$l_i:(StartPt_i, EndPt_i),\ (i=1,2,3)$$

and their corresponding model lines are

$$l'_i:(StartPt'_i, EndPt'_i),\ (i=1,2,3).$$

Performing projection transformation on line $l'_i(i=1,2,3)$, we can get $l''i$ $(i=1,2,3)$,

$$l''_i:(StartPt''_i, EndPt''_i),\ i=(1,2,3),$$

where

$$StartPt''_i=\left(\frac{fStartPt'_i.x}{StartPt'_i.z+D_z}+D_x, \frac{fStartPt'_i.y}{StartPt'_i.z+D_z}+D_y\right),\ (i=1,2,3)$$

$$EndPt''_i=\left(\frac{fEndPt'_i.x}{EndPt'_i.z+D_z}+D_x, \frac{fEndPt'_i.y}{EndPt'_i.z+D_z}+D_y\right),\ (i=1,2,3).$$

The slant for line $l''_i(i=1,2,3)$ is $m''_i=\frac{EndPt''_i.y-StartPt''_i.y}{EndPt''_i.x-StartPt''_i.x}$, $(i=1,2,3)$, then the distance from point $StartPt_i(i=1,2,3)$ to line $l''_i(i=1,2,3)$ is

$$d_i=\frac{m''_iStartPt_i.x-StartPt_i.y-m''_iStartPt''_i.x+StartPt''_i.y}{\sqrt{m''^2_i+1}},\ (i=1,2,3)$$

$$d'_i=\frac{m''_iEndPt_i.x-EndPt_i.y-m''_iStartPt''_i.x+StartPt''_i.y}{\sqrt{m''^2_i+1}},\ (i=1,2,3).$$

With d_i, $d'_i(i=1,2,3)$ and Eq. (12), we can get one solution of vector $(\triangle D_x,\triangle D_y,\triangle D_z,\triangle\varphi_x,\triangle\varphi_y,\triangle\varphi_z)$, then we can adjust vector $(D_x,D_y,D_z,\varphi_x,\varphi_y,\varphi_z)$ with Newton's method until d_i and $d'_i(i=1,2,3)$ are less than a certain threshold. Finally the projection parameters of 3D human model are reached.

2.2 Conversion from projection parameters to human motion model

When the calculation of projection parameters for every part of human model is finished, we need to convert these parameters to fit the skeleton structure of human model, then spatial parameters of the vectors that correspond to each part of the human model can be calculated. Thus the conversion from projection parameters to human motion model is completed.

2.3 The implementation of projection parameters controlled human motion

Based on the above idea, we have developed a video-based animation system called Video&Animation Studio. Figures 1(a)(b) to 2(a)(b) show a human walking example implemented by Video&Animation Studio. The figures on the left side are video data and a set of skeleton features defined by us, while the figures on the right side are the motion of the human model which is controlled by the sets of skeleton features in the video.

(a) (b)

Fig. 1

© 中国科学院软件研究所 http://www.jos.org.cn

(a)

(b)

Fig. 2

3 Conclusions

From the experimental result, we can see that the result of this method is satisfying. But there are four points that should be noted: 1. The choice of initial values for projection parameters is very important. Actually we can get a set of more accurate initial values by using some interactive methods. 2. The choice of feature lines in video is also very important. They should be the spline of video human. 3. As for the error in Newton's method, we'd like to regard the body as a whole. The error of a specific part of the body can be somehow unsatisfying, but the error of the whole body should be the minimum. 4. As for the processing of video, more detailed description can be found in Ref. [8].

References

1 Gibson J J. The Ecological Approach to Visual Perception. Houghton-Mifflin. Boston, MA, 1979

2 Marr D. Vision (Freeman, San Francisco, 1982)

3 Barnard S T. Interpreting perspective images. Artificial Intelligence, 1983,21:435～462

4 Barrow H G, Tenenbaum J M. Interpreting line drawings as three-dimensional surfaces. Artificial Intelligence, 1981,17:75～116

5 Brooks R A. Symbolic reasoning among 3-D models and 2-D images. Artificial Intelligence, 1981,17:285～348

6 Hochberg J E, Brooks V. Pictorial recognition as an unlearned ability: a study of one child's performance. American Journal of Psychology, 1962,75:624～628

7 Lowe D G. Solving for the parameters of object models from image descriptions. In: Proceedings ARPA Image Understanding Workshop, College Park, MD, 1980. 121～127

8 Lao Zhi-qiang. Study on imaginary-based intelligent animation techniques [Ph. D. Thesis]. Zhejiang University, 1997
(劳志强. 基于形象思维的智能动画技术的研究[博士学位论文]. 浙江大学,1997)

基于模型的人体动画方法

劳志强　潘云鹤

(浙江大学 CAD&CG 国家重点实验室　杭州　310027)

(浙江大学人工智能研究所　杭州　310027)

摘要　文章从点点对应,线线对应入手,首先给出了根据已知二维影像特征求得与其对应的三维模型特征参数的方法,然后给出了已知人体二维影像特征求解三维人体模型特征参数方法的详细描述.基于这一思想,实现了一个动画系统 Video & Animation Studio.文章最后给出了一个由该系统实现的人体行走的例子.

关键词　人体动画,影像特征和模型特征的对应关系.

中图法分类号　TP391

© 中国科学院软件研究所 http://www.jos.org.cn