MULTIPLE CURVATURE BASED APPROACH TO HUMAN UPPER BODY PARTS DETECTION WITH CONNECTED ELLIPSE MODEL FINE-TUNING

Richard Yi Da Xu, Michael Kemp
School of Computing and Mathematics
Charles Sturt University, Australia
{rxu, mkemp}@csu.edu.au

ABSTRACT

In this paper, we discuss an effective method for detecting human upper body parts from a 2D image silhouette using curvature analysis and ellipse fitting. First we smooth the silhouette so that we can determine just the global features: the head, hands and armpits. Next we reduce the smoothing to detect the local features of the neck and elbows. We model the human upper body by multiple connected ellipses. Thus we segment the body by the extracted features. Ellipses are fitted to each segment. Lastly, we apply a non-linear least square method to minimize the differences between the connected ellipse model and the edge of the silhouette.

Index Terms— Pose recognition, contour, ellipse fitting

1. INTRODUCTION

Pose recognition problems have long been studied in many computer vision and pattern recognition literature and employ a variety of features and modeling methods. Some of the features used include edges [1, 2], foreground silhouette [3, 4], motion histories [5] and optical flow. The modeling methods include the use of template-based distance measures, such as Chamfer Distance [1] and parameterized approaches, such as Mixture Density Models [6].

Pose detection (or initialization) is typically performed on an initial video frame followed by pose tracking where the pose parameters obtained from the current frame is used as a starting value for the subsequent video frames.

In this paper, we propose an effective pose detection method: We model the human upper body using multiple connected ellipses with its initial guess derived from the curvature analysis technique. The model state representation has a finite dimension, and enjoys a continuous solution space. This is in contrast to the previous parameterized approach, where only a finite number of predefined pose classes are permitted. A recent representative example is found in [6], where the authors use Generalized Expectation Maximization (GEM) to assign edge pixels to body parts and to find the body pose that maximizes the likelihood of the resultant assignments. However, in the Maximization (M)-step of the algorithm, a "better" pose parameter is selected from the predefined data sets to increase the likelihood probability. The use of only discrete number of classes can result in misalignment between the image data and the model, especially when the number of predefined models parameters is small. On the other hand, our approach does not use predefined pose classes, but considers an infinite number of possible human upper body representations. The only constraint imposed is the connectivity between the body limbs in the form of connected ellipses. In addition, the initial pose guess used in our work is based on a curvature analysis, experiment shows that it outperforms other silhouette based pose descriptors such as geodesic distance [3]. The combined advantages make our algorithm more suitable as a candidate for complex temporal gesture analysis such as view independent action recognition frameworks.

The rest of this paper describes our method in detail: we begin by using curvature analysis at different levels of smoothing to extract key features of the body (head, hands, neck and elbows) (section 2). Then we use these extracted features to give an initial fit for our multiple connected ellipse model (section 3). Lastly, we use the Levenberg-Marquardt algorithm to fine-tune the fit. In section 4, we also discuss our results and making comparisons to the existing methods.

2. SILHOUETTES CURVATURE ANALYSIS

2.1. Preprocessing

Human silhouette information can often be made available using the off-the-shelf background subtraction algorithms [7]. It allows us to obtain a set of foreground blobs, such as the ones shown in figure 1.

Fig 1: images of human upper-body foreground blob

Background subtraction picks up noise (typically caused by shadows) as well as the human silhouette. We remove the

noise by using the connected component containing the centroid. We then determine the blob's contour from the blob's bottom left to its bottom right. This is shown in figure 2, (the green and blue points indicate respectively the beginning and end of the contours).

Fig 2. result of human upper-body silhouettes pre-processing

2.2. Obtain silhouette's global features

Closely associated with image silhouette processing, are the curvature-based analysis [4, 8] methods, which use the 2D curvature function or *kappa* function:

$$\kappa(u) = \frac{\dot{x}(u)\ddot{y}(u) - \dot{y}(u)\ddot{x}(u)}{(\dot{x}(u)^2 + \dot{y}(u)^2)^{3/2}} \qquad (1)$$

Curvature is more invariant to translation, scaling and rotation compared to other methods that employ point locations features, such as the geodesic distance to its centroid [3]. Thus curvature is widely used for object recognition applications [8].

For an ideal curve, curvature is minimal at extremities for hands and head and maximum at armpits and necks. Between each maxima and minima, the curvature crosses through zero. Therefore the number of zero-crossing characterizes the shape of the curve.

The contours we obtain initially have many zero-crossings: some through noise and others due to finer detail of the silhouette. In order to obtain the global features of the human upper body (hands, head and trunk) we need eight zero-crossings. We achieve this by smoothing the contour a sufficient amount. The greater the smoothing, the lower the number of zero-crossing [8]. We smooth the original contour $\mathbf{X}(u)$ by convolving it with a Gaussian smoothing kernel to obtain $\mathbf{X}_{smooth}(u)$:

$$\mathbf{X}_{smooth}(u) = \mathbf{X}(u) \otimes g(u, \sigma) \qquad (2)$$

where $g(u,\sigma)$ is the Gaussian kernel with standard deviation σ and $\otimes$ is the convolution operator. An example is shown in figure 3, where figure 3(a) is the original curve $\mathbf{X}(u)$, and figure 3(c) is the smoothed curve $\mathbf{X}_{smooth}(u)$.

Different values of the smoothing parameter σ result in different kappa functions. For example, figure 3(b) is the plot of kappa function for $\mathbf{X}(u)$ and figure 3(d) is the plot of kappa function for $\mathbf{X}_{smooth}(u)$.

Thus the goal is to find a value $\sigma_{optimal}$ which smoothes the sufficient amount. To this end we collected a database of human silhouettes. For each human silhouette, we recorded an interval of σ values that generates eight zero crossings for the kappa function after convolution. We defined the middle value of this interval to be $\sigma_{mid,i}$ and its halfway distance to be $\sigma_{range,i}$.

We assign the initial $\sigma_{optimal}$ to be the mean of $\{\sigma_{mid,i}\}$, and the searching range σ_{range} to be the mean plus the standard deviation of $\{\sigma_{range,i}\}$. The search direction depends on the number of zero crossings that the current $\sigma_{optimal}$ generates, i.e, we increase $\sigma_{optimal}$ value when the number of zero crossings present is excessive, and decrease it otherwise. If we are unable to obtain the required number of zero-crossings within σ_{range}, the search for the human pose within the current video frame stops.

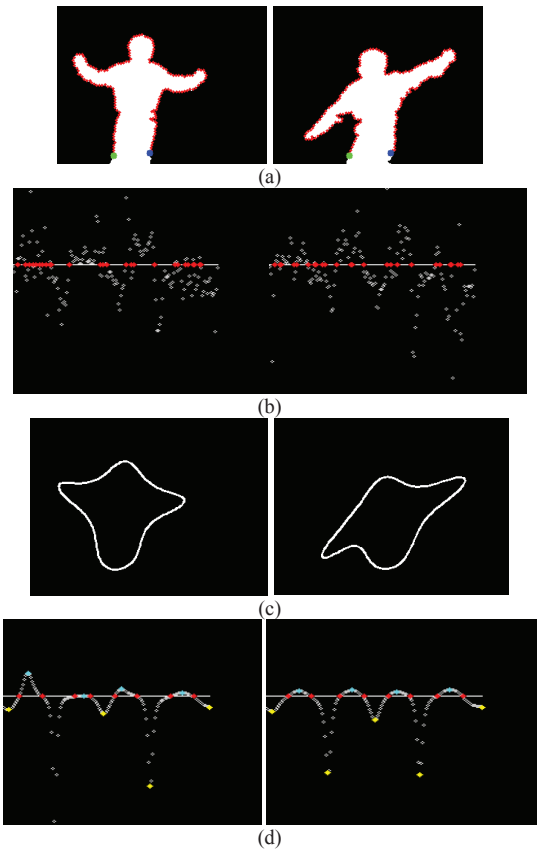

Fig 3: (a) is the original curve $\mathbf{X}(u)$ overlaid on the silhouette and (b) is its kappa function. (c) is $\mathbf{X}_{smooth}(u)$ and (d) is its corresponding kappa function. In (b) and (d), the red dots indicate the kappa zero-crossings. In (d), the cyan and yellow dots represent kappa maxima and minima respectively.

2.3. Fine-tuning to the local features

After obtaining the required eight zero crossings in the kappa function, we plot the three middle minima kappa values in $\mathbf{X}(u)$. The fourth minimum is split into two as the left and right-most minima in figure 3(d) (which corresponds to the middle of the trunk), and is not used further. Also we plot the four maximum kappa values in $\mathbf{X}(u)$ as shown in figure 4a. It can be seen that three minima generally correspond to the top of the head and tips of hands, and two of the maxima generally correspond to the armpits. However, the neck positions do not correspond reliably with its corresponding maxima. In addition, the

elbow position is also difficult to extract from the kappa function consistently.

In order to obtain the local neck positions more accurately, we consider just the original $\mathbf{X}(u)$ segment which starts and ends at the two corresponding maxima of $\mathbf{X}_{smooth}(u)$. We smooth this segment with a lower sigma value, typically $\sigma = 2.8$ (refer to figure 4). We then refine the neck positions to be the two maxima of the resultant kappa function.

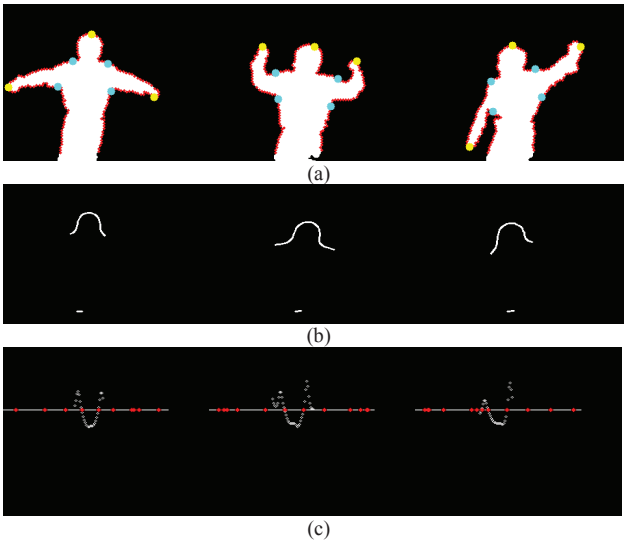

Fig 4: (a) plot of the kappa function maxima and minima to the original silhouette, showing incorrect neck positions (b) smoothed contour segment (using lower sigma) between the two shoulder maxima (c) new kappa function for the segment, showing two closer maxima

In order to obtain the four elbow positions, we use the 2D arc-length function: $\int_0^v \sqrt{\dot{x}^2 + \dot{y}^2}\, du$. The lower elbow position is defined to be half of the arc-length between the kappa maxima for the armpit and kappa minima for the hand. The rest of the four elbow points are obtained similarly. We then segment the body using the extracted features. In figure 5, we show each of the obtained body parts on the original $\mathbf{X}(u)$ using different colours.

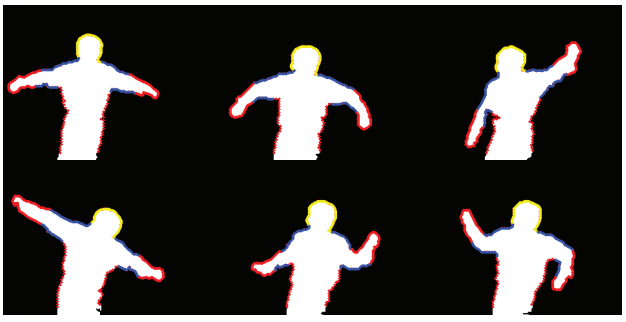

Fig 5: Segments of a human upper body silhouette: head, body and upper and lower arms.

3. HUMAN UPPERBODY CONNECTED ELLIPSE STRUCTURE FITTING

Once the different segments of a human upper body are obtained, we then fit a connected ellipse model to the silhouette. The connected ellipse model used in our work is comprised of six ellipses and three joints, shown in figure 6.

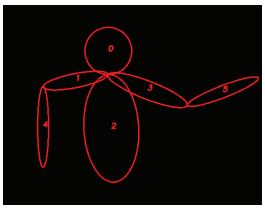

Fig 6: the connected ellipse model for a human upper body, showing the ellipse numbers and joints structure

3.1. Individual ellipse fittings

After step 2.3, contour points of each segment are individually fitted to an ellipse using least square methods. The results are shown in the left column of figure 7. Note that at this stage, the ellipses are not connected.

In order to match our model, we need to connect the ellipses. For each joint node, we realign its position to the average of nearest major axis points over all the ellipses connected at that joint. Major axis points that are not connected at any joint are not altered. We also note that the head ellipse is usually fitted better than others. Therefore, as an exception, at the joint point which involve the head ellipse, no averaging is required, all other ellipse's major axis points simply realign to coincide with the head ellipse major axis point.

3.2. Final ellipse fine-tuning using Levenberg-Marquardt

We further improve our results for a closer fit between the ellipses and the silhouette's edges by using the Levenberg-Marquardt non-linear least square method.

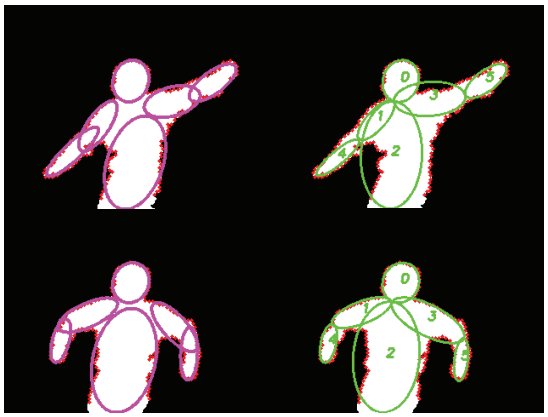

Fig 7: ellipse fitting result, the magenta ellipse are the unconnected ones and the green are the connected ellipses

Firstly, we define the distance between a point (x_j, y_j) to the closet edge of an individual ellipse (which is centered at (0,0), with semi-axes lengths of a and b) to be:

$$d(\text{ellipse},(x_j,y_j)) = \sqrt{x_j^2+y_j^2}\left|1-\frac{1}{\sqrt{(x_j/a)^2+(y_j/b)^2}}\right| \quad (3)$$

A similar function is also defined in [9]. To obtain the distance for a general ellipse, we use an orthogonal transformation to center it at (0,0) and align the axes with coordinate axes. Once we obtained the distance function of an individual ellipse, the proximity between a contour point (x_j, y_j) to the edge of the entire ellipse tree is defined as:

$$prox_j(\mathbf{p}) = \exp\left(-\sum_{i=1}^{N_{Ellipse}} \exp\left(-d(\text{ellipse}_i(\mathbf{p}),(x_j,y_j))\right)\right) \quad (4)$$

p is the joint ellipse parameter vector. The Levenberg-Marquardt (L-M) algorithm is then iteratively applied to find a solution for **p** that minimizes:

$$\tfrac{1}{2}\sum_{j=1}^{M}\left(prox_j(\mathbf{p})\right)^2 \quad (5)$$

for the *M* silhouette contour points. Therefore, an ellipse set is obtained with parameter **p** which has a closer match to the silhouette edge points. The results are shown in figure 8. Notice that the blue ellipse sets which are the result of L-M, match the contour more precisely than the green ellipse sets from section 3.1.

4. DISCUSSION AND RESULTS

In this paper, we presented a set of methods for detecting human upper body parts. We first used curvature analysis to extract the global and local silhouette features. We then fitted six individual ellipses to each contour segment and connected them according to a predefined ellipse joints structure. We further improved the results for a closer fit between the ellipses and the silhouette's edges using the LM non-linear least square method. The blue ellipses in Figure 8 show the results for four different human poses. This method has been implemented for real-time usage as well. A demonstration is available in video format on Youtube:

http://www.youtube.com/watch?v=kU7TmByKJoI

Real-time performances were achieved when the final LM fine-tuning (section 3.2) is skipped; in that case the system is able to operate at 20 fps on a quad core PC. The processing time increases dramatically when LM fine-tuning is applied, which reduces the frame rate to 3fps, with 80 sampled edge points used and L-M is capped at 100 iterations. However, as shown in section 3.2, the ellipses are fitted to the silhouette edge more accurately using the L-M step, which is a tradeoff between performance and efficiency.

In our work, the iterative LM step varies the pose parameters continuously until convergence is achieved. In comparison in [6], where in order to ensure a valid pose is obtained, the final solution is simply sampled from a database of known poses to increase the likelihood of matching. Since our method uses a continuous scale, closer alignment between the model and silhouette is possible. Also since arbitrary sized jumps can be made, it is possible to get quicker convergence. The advantages will be more obvious when the number of predefined poses become less in number.

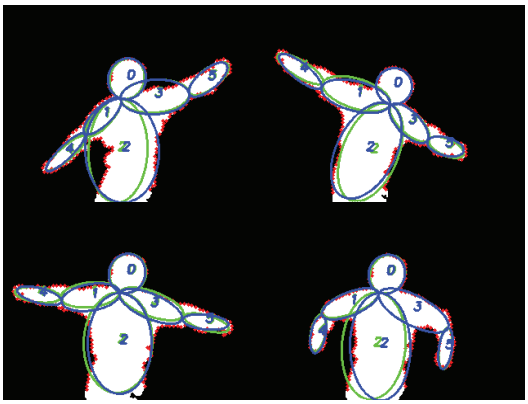

Fig 8: ellipse fine-tuning result, green ellipses are results from section 3.1 and the blue ones are the results after L-M in section 3.2

In terms of body parts extraction, the curvature analysis method with varying sigma employed in our method outperforms other silhouette based pose descriptors, such as geodesic distance [3]. Although the author in [3] has not explicitly indicated how they achieved accurate maxima detection. Using the 40 key pose images captured with varying poses, our algorithm can accurately extract head and other body parts in 39 images. Using the geodesic approach [3] with additional Gaussian smoothing, only 34 were accurate in head and hands detection and 25 were accurate in arm pits detection.

REFERENCES

1. Gavrila, D.M., *A Bayesian, Exemplar-Based Approach to Hierarchical Shape Matching*. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007. **29**(8): p. 1408-1421.
2. Dimitrijevic, M., V. Lepetit, and P. Fua, *Human body pose detection using Bayesian spatio-temporal templates* Computer Vision and Image Understanding, 2006. **104** (2): p. 127 - 139.
3. Correa, P., et al. *Silhouette-based probabilistic 2D human motion estimation for real-time applications*. in *IEEE International Conference on Image Processing, 2005. ICIP 2005*. 2005.
4. Chang, C.-C., I.-Y. Chen, and Y.-S. Huang. *Hand Pose Recognition Using Curvature Scale Space*. in *16th International Conference on Pattern Recognition (ICPR'02)*. 2002.
5. Bradski, G.R. and J.W. Davis, *Motion segmentation and pose recognition with motion history gradients*. Machine Vision and Applications, 2002. **13**: p. 174–184.
6. Fossati, A., et al. *Tracking Articulated Bodies using Generalized Expectation Maximization*. in *CVPR Workshop on Non-Rigid Shape Analysis and Deformable Image Alignment*. 2008. Anchorage, USA.
7. Stauffer, C. and W.E.L. Grimson. *Adaptive background mixture models for real-time tracking*. in *IEEE CVPR*. 1999.
8. Mokhtarian, F., S. Abbasi, and J. Kittler, *A new approach to computation of curvature scale space image for shape similarity retrieval*, in *Image Analysis and Processing*. 1997. p. 140-147.
9. Jeune, F.L., et al., *Tracking Of Hand's Posture And Gesture*. 2004, CERTIS, ENPC, http://certis.enpc.fr/publications/papers/04certis02.pdf.