Recognizing Posture in Pictures with Successive Convexification and Linear Programming

Hao Jiang, Ze-Nian Li and Mark S. Drew
School of Computing Science, Simon Fraser University
Vancouver, BC, V5A 1S6, Canada

Abstract

We present an image matching method for recognizing human postures in cluttered images and videos. A novel “successive convexification” scheme is developed for matching body postures. Using local image features, the proposed scheme is able to accurately locate and match human objects over large appearance changes. Postures are recognized based on similarity measures between exemplars and located target objects. Experiments show very promising results for the proposed scheme in recognizing and detecting human body postures in images and videos.

Keywords: Human Posture Recognition, Pattern Matching, Linear Programming, Successive Convexification.

1 Introduction

Recognizing human posture in images and videos is an important task in many multimedia applications, such as multimedia information retrieval, human computer interaction, and surveillance. Posture is a snapshot of human body configuration. A sequence of postures can be combined together to generate meaningful gestures. In many cases, a posture in one single image also conveys meaningful information. For example, it is possible for a human observer to disambiguate actions such as walking, running, standing, sitting, etc., from just a single image. In recent years, recognizing human body postures in images or videos with a good deal of confounding background clutter has received much interest.

In this article, we present a posture detection method based on local image features and successive convexification image matching [21] [22] [23]. Image matching based on successive convexification operates very differently from previous methods such as Relaxation Labeling (RL) [1], Iterative Conditional Modes (ICM) [2], Belief Propagation (BP) [3], Graph Cut (GC) [4], and other

convex programming based optimization schemes [5] [6] [7]. The proposed scheme represents target points for each template point with a small basis set. Successive convexification gradually shrinks trust region for each template site and converts original hard problem into a sequence of much simpler convex programs. This greatly speeds up searching, making the method well suited for large scale matching and posture recognition problems. In experiments, we show successful application of the proposed scheme in detecting human postures and actions in cluttered images and video sequences.

1.1 Related Work on Posture Recognition

Recognizing human body configuration in controlled environments has been intensively studied in many experimental and commercial systems; to name a few: MIT Media Lab’s KIDSRROOM [8], ALIVE [9], Emering et al.’s gesture recognition system [10] and Vivid Group’s gesture recognition system [11] aimed at HCI applications. These systems rely on segmentation of human objects from the background in a specific, restricted environment (the KIDSRROOM, ALIVE, Vivid group’s system etc.) or by position/velocity sensors attached to human subjects [10]. To facilitate the segmentation process, other systems use infrared cameras [12] or multiple cameras [13]. These systems are more expensive to deploy than simple monocular visible-light camera systems.

In uncontrolled environments, recognizing human body postures becomes a challenging problem because of background clutter, articulated structures of human bodies, and large variability of clothing. To overcome these difficulties, different methods based on directly matching *templates* to targets have been studied. One method is to detect human body parts [14] [15] [16] and their spatial configuration in images as illustrated in Fig. 1 (a). Body-part methods only involve a few templates to represent each body part. The shortcoming of this method is that body parts are difficult to locate in many uncontrolled cases, mainly due to clothing changes, occlusion, and body-part deformation. Currently, body-part based schemes are used for recognizing relatively simple human postures such as walking [15] and running [17]. Another method recognizes human postures based on small local image features. As illustrated in Fig. 1 (b), this scheme matches postures as whole entities and does not distinguish body parts explicitly. In this article, we follow this scheme. Most previous methods based on matching local image features [19] [20] assume a relatively clean background. When background clutter increases, distinguished features are weakened and simple matching schemes cannot generate desirable results. The successive convexification based scheme we now outline presents a method to robustly and efficiently solve the problem.

2 Posture Recognition as a Matching Problem

Posture recognition is inherently an image matching problem. After matching posture template to target object, we can compare their similarity and carry out posture recognition. Posture matching,

Figure 1: Posture recognition by matching: (a): body-part based; (b): matching a whole entity, using local image features.

can be stated as an energy minimization problem:

$$\min \{E_{Matching} + \lambda \cdot E_{Smooth}\} \quad (1)$$

We would like to find an optimal matching from template feature points to target points. The goal is to minimize the matching cost, the first term above, and at the same time smooth the matching with the second, regularity (or “smoothness”) term. The multiplier λ balances the matching cost and the smoothness term.

In this article, the energy minimization problem is formulated based on Eq. (1) as

$$\min_{\mathbf{f}} \left\{ \sum_{s \in S} c(\mathbf{s}, \mathbf{f}(\mathbf{s})) + \lambda \sum_{\{\mathbf{p}, \mathbf{q}\} \in \mathcal{N}} d(\mathbf{f}(\mathbf{q}) - \mathbf{f}(\mathbf{p}), \mathbf{q} - \mathbf{p}) \right\}, \quad (2)$$

where S is the feature point set; $\mathcal{N}$ is the neighboring point set; $\mathbf{f}(\mathbf{s})$ maps 2D point $\mathbf{s}$ in template image to a 2D point in target image; $c(\mathbf{s}, \mathbf{f}(\mathbf{s}))$ is the cost of matching target point $\mathbf{f}(\mathbf{s})$ to $\mathbf{s}$ (e.g., our block-based image measure below); $d(\cdot, \cdot)$ is a distance function. We focus on the problem where $d(\cdot, \cdot)$ is the city block distance. The smoothness term enforces that neighboring template points should not travel too far from each other, once matched. There are different ways to define the neighbor pair set. One natural way is to use a Delaunay triangulation over the feature points in the template, and identify any two points connected by a Delaunay graph edge as *neighbors*.

Fig. 2 illustrates the matching problem. In Fig. 2, points $\mathbf{p}$ and $\mathbf{q}$ are two neighboring template feature points and their targets are $\mathbf{f}(\mathbf{p})$ and $\mathbf{f}(\mathbf{q})$ respectively. Intuitively, we should minimize

Figure 2: Matching postures.

the matching costs and at the same time try to make the matching consistent by minimizing the difference of vectors $q - p$ and $f(q) - f(p)$.

2.1 Features for Matching

For posture recognition problems, the features selected for the matching must be insensitive to appearance changes of human objects. The edge map contains most of the shape information for an object, and at the same time is not very sensitive to color changes. Edge features have been widely applied in Chamfer (edge-based) matching [18] and shape context [25] matching. We have found that small blocks centered on the edge pixels, of a distance transform image are expressive local features. Here, a *Distance Transform* converts a binary edge map into a corresponding grayscale representation, with the intensity of a pixel proportional to its distance to the nearest edge pixel. To incorporate more context information, we can further applied a log-polar transform to a distance transform image [23]. The matching cost can then be represented as the normalized mean absolute difference between these local image features. Local image features are not reliable in image matching, and therefore a robust matching scheme as is presented in the following is required.

2.2 Linear Programming Matching

The energy optimization problem in Eq. (2) is usually nonlinear and highly non-convex, i.e., it has many local minima. Such problems are difficult to solve without a good initialization process. Instead of trying to optimize the problem directly, we convert it into an approximated linear programming (LP) problem [22], [21] and [23].

The basic idea is that we introduce weights which can be interpreted as a set of (float) *soft decisions* for matching target points to template feature points. A target point can then be represented as the linear combination of *representative* target points that we call the *basis target points*. The cost of matching is *approximated* as the weighted sum of costs of these basis points. Finally the

Figure 3: Lower convex hull. Left: a cost surface; Middle: Lower convex hull facets; Right: The label basis $\mathcal{B}_s$ contains coordinates of the lower convex hull vertices (Solid dots are basis points).

Sidebar 1: Properties of LP Formulation

The LP formulation has several interesting properties:

1. *For general cost function, the linear programming formulation solves the continuous extension of the reformulated matching problem, with each matching cost surface replaced by its lower convex hull.*
2. *The most compact basis set contains the vertex coordinates of the lower convex hull of the matching cost surface.*

By this property, there is no need to include all the matching costs in the optimization: we need only include those corresponding to the basis target points. This is one of the key steps to speed up the algorithm.

3. *If the convex hull of the cost function is strictly convex, nonzero weighting basis labels must be “adjacent”.* Here “adjacent” means the convex hull of the nonzero weighting basis target points cannot *contain* other basis target points.
4. *If we solve the linear programming problem by the simplex method, there will be at most 3 nonzero-weight target points for each feature point in the template.*

The optimization is reduced to just a fast descent through a few triangles in the target point space for each site.

smoothness term is linearized by using auxiliary variables [24].

In some special cases, this linear program can be used to exactly solve the continuous extension of the matching problem; in general situations, it is an approximation of the original problem. Sidebar 1 lists some properties [22] of the LP. Fig. 3 illustrates a cost surface, its lower convex hull and the basis target points.

Figure 4: Object Matching using successive convexification.

After the convexification process, the original non-convex optimization problem turns into a convex problem and an efficient linear programming method can be used to yield a global optimal solution for the approximation problem.

2.3 Successive Convexification

Because of the convexification effect of the linear programming relaxation, the approximation is coarser for larger search region in the target image. Thus the LP solution will be more precise if we can narrow down the searching range. A successive relaxation scheme is thus proposed to solve the coarse approximation problem. We construct linear programs recursively, based on the previous searching result, and gradually shrink the trust region for each site, systematically. But note that we *convexify the original cost function* again (i.e., we “re-convexify”) in the smaller region. Fig. 4 shows the procedure.

Anchors are used to control the trust regions. To locate anchors, a consistent rounding process [22] is applied to LP solution of the previous stage. The new trust region for each site is a smaller rectangular region that contains the anchor, for example, a region centered on the corresponding anchor. Example 1 illustrates the successive convexification procedure for a simple 1D matching problem.

Example 1 (A 1D problem): Assume there are two sites $\{1, 2\}$ and for each site the target point set is $\{1..7\}$. The objective function is $\min_{\{\rho_1, \rho_2\}} [c(1, \rho_1) + c(2, \rho_2) + \lambda |\rho_1 - \rho_2|]$. In this example we assume the matching costs are $\{c(1, j)\} = [1.1, 6, 2, 7, 5, 3, 4]$, $\{c(2, j)\} = [5, 5, 5, 1, 5, 1, 5]$; and $\lambda = 0.5$.

Based on the proposed scheme, the problem is solved by the sequential LPs: $\mathcal{LP}_0$, $\mathcal{LP}_1$ and $\mathcal{LP}_2$.

- In $\mathcal{LP}_0$ the trust regions of sites 1 and 2 are both $[1, 7]$. Constructing $\mathcal{LP}_0$ based on the proposed scheme corresponds to solving an approximated problem in which $\{c(1, j)\}$ and

Figure 5: An example of successive convexification matching.

$\{c(2, j)\}$ are replaced by their lower convex hulls respectively (see Fig. 5). Step $\mathcal{LP}_0$ uses basis labels $\{1, 6, 7\}$ for site 1 and basis labels $\{1, 4, 6, 7\}$ for site 2. $\mathcal{LP}_0$ has solution $\xi_{1,1} = 0.4$, $\xi_{1,6} = 0.6$, $\xi_{1,7} = 0$, $\rho_1 = (0.4 * 1 + 0.6 * 6) = 4$; and $\xi_{2,4} = 1$, $\xi_{2,1} = \xi_{2,6} = \xi_{2,7} = 0$, $\rho_2 = 4$. Based on the rules for anchor selection [22], we fix site 2 with $\mathcal{LP}_0$ solution 4, and search for the best target point for site 1 in the region $[1, 7]$ using the non-linear objective function; we get the anchor 3 for site 1. Using similar method for site 2, we get its anchor 4.

- Further, the trust region of $\mathcal{LP}_1$ is $[1, 5] \times [2, 6]$ by shrinking the previous trust region diameter by factor of 2. The solution of $\mathcal{LP}_1$ is $\rho_1 = 3$ and $\rho_2 = 4$. The new anchor is 3 for site 1 and 4 for site 2.
- Based on $\mathcal{LP}_1$, $\mathcal{LP}_2$ has new trust region $[2, 4] \times [3, 5]$ and its solution is $\rho_1 = 3$ and $\rho_2 = 4$. Since 3 and 4 are the anchors for site 1 and 2 respectively and in the next iteration the diameter shrinks to unity, the iteration terminates. It is not difficult to verify that the configuration $\rho_1 = 3$, $\rho_2 = 4$ achieves the global minimum.

Interestingly, for the above example ICM or even the Graph Cut only finds a local minimum, if initial values are not correctly set. For ICM, if ρ_2 is set to 6 and the updating is from ρ_1 , the iteration will fall into a local minimum corresponding to $\rho_1 = 6$ and $\rho_2 = 6$. The Graph Cut scheme based on α -expansion will have the same problem if the initial values of both ρ_1 and ρ_2 are set to 6.

Example 2 (An 2D problem): Fig. 6 illustrates an example for matching a triangle in clutter using successive convexification. The trust region updating and convexification process for two points on the template are illustrated. The black rectangles in Figs. 6 (d), (e) and (f) indicate the

Figure 6: Object matching in cluttered image.

trust regions for the two selected points in three successive LP stages. The convexified matching cost surfaces for each site in these trust regions are illustrated in the second row of Fig. 6. These convex surfaces are supported by a very small number of vertices corresponding to the basis target points. The 3-stage successive convexification scheme locates the target in clutter accurately.

With a simplex method, an estimate of the average complexity of successive reconversion linear programming is $O(|S| \cdot (\log |L| + \log |S|))$, where S is the set of template feature points and L is the target point set. Experiments also confirm that the average complexity of the proposed optimization scheme increases more slowly with the size of target point set than previous methods such as Belief Propagation, whose average complexity is proportional to $|L|^2$.

2.4 Measuring Similarity

After posture matching from a template to target object, we need to decide how similar these two constellations of matched points are and whether the matching result corresponds to the same posture as in the exemplar. We use the following quantities to measure the difference between the template and the matching object.

We first define measure D as the average pairwise length changes from the template to the target. To compensate for the global deformation, a global affine transform is first estimated based on the matching and then applied to the template points before calculating D . D is further normalized with respect to the average edge length of the template. The second measure is the average feature matching cost M . The matching score is simply defined as the linear combination of D and M . Experiments show that only about 100 randomly selected feature points are needed in calculating D and M .

The above posture matching method can also be extended to matching video sequences to detect actions [23] by introducing a center continuity constraint. In the following, we present experimental results of posture and action detection in images and videos.

3 Experimental Results

In this section, we first compare the proposed matching scheme with BP and ICM using synthetic ground truth data. Then we show experiments to test the proposed human posture detection scheme using real video sequences.

3.1 Matching Random Dots

In this experiment we compare the performance of successive convexification linear programming (SCLP) with BP and ICM for binary object detection in clutter. In our experiments, the templates are generated by randomly placing 100 black dots into a 128×128 white background image. A 256×256 target image is then synthesized by randomly translating and perturbing the block dot positions from those in the template. Random noise dots are then added to the target image to simulate background clutter. For each testing situation we generate 100 template and target images. In this experiment, we match the graylevel distance transformation of the template and target images. Fig. 7 compares results using the proposed matching scheme with using BP and ICM. The histograms show error distributions of different methods. In this experiment, all the methods use the same energy function. SCLP has similar performance to BP and much better than the greedy ICM scheme in cases of large distortion and cluttered environments. SCLP is much more efficient than BP when the number of target points exceeds 100. With a 2.8GHz PC, for a matching problem with 80 template points and 1000 target candidate points, SCLP has an average matching time 10 secs with 4 iterations, while BP takes about 100 secs for just one iteration.

3.2 Finding Postures in Video

Finding postures in video sequences using exemplar postures is a very useful application. We first test the method with a “yoga” sequence, which is about 30-min long. We choose three different posture exemplars from another section of the video. By specifying the region of interest, graph templates are automatically generated from the exemplars. Each template is then compared with video frames in the test video. The shortlists based on their matching scores are shown in Figs. 8 (b, c, d). The templates are shown as the first image in each shortlist. The Recall-Precision curves are displayed in Fig. 12 (a).

Fig. 9 illustrates the performance of the proposed scheme in matching objects with *large appearance differences*. We use a flexible toy as the template object and search in video sequences for similar postures of actual human bodies. Two sequences are used in testing: the first, shown in Fig. 9, has 500 frames and the other has 1,000 frames. There are fewer than 10% true targets in the video sequence. The vertical and horizontal edges in the background are very similar to the edge features on human bodies, and this presents a major challenge for object location and matching. The shortlists of matching results are shown in Figs. 9 (b, d), and Recall-Precision curves are shown in Fig. 12 (b).

Figure 7: Histogram of matching errors using SCLP, BP and ICM.

In another experiment, we search a figure skating sequence about 30-min long to locate similar postures as exemplar postures. The figure skating program contains 5 skaters, with quite different clothing. The audience in the scene presents strong background clutter, which may cause problems for most matching algorithms. The sampling rate for the video is 1 frame/second. Fig. 10 shows shortlists of posture searching based on the matching scores for three different postures. The templates are shown as the first image in each shortlist. The Recall-Precision curves are shown in Fig. 12 (c).

In the previous experiments, we search for postures in videos that contain a single object in each video frame. In this experiment, we consider posture recognition for videos that may contain multiple objects in each frame. We would like to locate objects with specific postures in hockey games. Hockey is a fast paced game, with fast player movements and camera motion. Detecting activities of hockey players is an interesting and challenging application. The background audience and patterns on the ice also make posture recognition a hard problem. To deal with multiple targets in images, we apply composite filtering first. The composite template is constructed as the average of 200 randomly selected hockey players. To reduce the influence of clothing, these images are converted to distance transformed images for composite template construction and composite filtering. For each input video frame, the positions of local valleys of the composite filter residue image are potential object centers. Rectangular image patches centered on these object centers are

Figure 8: Matching human postures in yoga sequence.

Figure 9: Matching human postures using flexible toy object template.

Figure 10: Figure skating posture detection.

cut from each video frame and forwarded to linear programming detail matching to compare their similarity with the posture template. Fig. 11 (a) shows the shortlist of searching for a shooting action in a 1000-frame video sequence. Two instances of the shooting action are successfully detected at the top of the shortlist. Fig. 11 (b) shows another posture detection result, for a 1000-frame video with another posture template. The shortlist of video frames and hockey players are shown, based on the matching scores. The matching score for a video frame is defined as the smallest object matching score in the frame. The Recall-Precision curves are shown in Fig. 12 (d).

We also compare Chamfer matching with the proposed scheme for posture detection. Fig. 13 shows the figure skating posture detection result using Chamfer matching. The template posture is the same as that of Fig. 10 (c). As shown in this result, Chamfer matching does not work well when there is strong clutter or large posture deformation.

3.3 Finding Activities in Videos

We further conducted experiments to search for a specific action in video using time-space matching [23]. An action is defined by a sequence of body postures. In these test videos, a specific action only appears a few times. The template sequence is swept along the time axis with a step of one frame, and for each instant we match video frames with the templates. Fig. 14 and Fig. 15 show experiments to locate two actions, kneeling and hand-waving, in indoor video sequences of 800 and 500 frames respectively. The two-frame templates are from videos of another subject in different environments. The videos are taken indoors and contain many bar structures which are

(a) Locating shooting posture in video with exemplar 1

(b) Locating postures in video with exemplar 2

Figure 11: Finding postures in hockey.

Figure 12: Recall-Precision curves.

Figure 13: Figure skating posture detection using Chamfer matching. The first image is the template image.

Figure 14: Searching “kneeling” in a 800-frame indoor sequence. (a) Templates; (b..g) Top 6 matches.

Figure 15: Searching “right hand waving” in a 500-frame indoor sequence. (a) Templates; (b..g) Top 6 matches.

very similar to human limbs. The proposed scheme finds all the 2 kneeling actions in the test video in top two of the short list; and all the 11 waving hand actions in the top 13 ranks. Fig. 16 shows the result of search for a “throwing” action in a 1500-frame baseball sequence. Closely interlaced matching results are merged and our method finds all the three appearances of the action at the top of the list. We found that false detection in our experiments is mainly due to similar structures in the background near the subject. Very strong clutter is another factor that may cause the matching scheme to fail. Prefiltering or segmentation operations to partially remove the background clutter can further increase the robustness of detection.

4 Conclusion

We have set out a novel posture detection method using successive convexification. This method is more efficient and effective than previous methods for posture matching in which a large target point set is involved. It can also solve problems for which other schemes fail. We use distance transforms of the edge maps to match the template and target images, and this representation facilitates matching objects with large appearance variations. Experiments show very promising

Figure 16: Searching “throwing ball” in a 1500-frame baseball sequence. (a) Templates; (b..g) Top 6 matches.

results for human body posture detection in cluttered environments.

By prefiltering video, confounding features can be partially eliminated from the target image, and matching will become more efficient, making it therefore possible to conduct real-time matching. Furthermore, dynamic models can also be incorporated to improve recognition accuracy. Finally, the proposed scheme has the potential to be directly applied to *general* object recognition problems.

References

- [1] A. Rosenfeld, R.A. Hummel, and S.W. Zucker, “Scene Labeling by Relaxation Operations,” IEEE Trans. Systems, Man, and Cybernetics, vol.6, no.6, pp.420–433, 1976.
- [2] J. Besag, “On the statistical analysis of dirty pictures”, J. R. Statis. Soc. Lond. B, 1986, Vol.48, pp.259–302.
- [3] Y. Weiss and W.T. Freeman. “On the optimality of solutions of the max-product belief propagation algorithm in arbitrary graphs”, IEEE Trans. on Information Theory, vol.47, no.2, pp.723–735, 2001.
- [4] Y. Boykov, O. Veksler, and R. Zabih, “Fast approximate energy minimization via graph cuts”, IEEE Trans. Pattern Analysis and Machine Intelligence, vol.23, pp.1222–1239, 2001.
- [5] J. Kleinberg and E. Tardos. “Approximation algorithms for classification problems with pairwise relationships: Metric labeling and Markov random fields”. In Proceedings of the 40th Annual IEEE Symposium on Foundations of Computer Science (FOCS’99), pages 14–23, 1999.
- [6] C. Chekuri, S. Khanna, J. Naor, and L. Zosin, “Approximation algorithms for the metric labeling problem via a new linear programming formulation”, Symp. on Discrete Algs. (SODA’01), pp.109–118, 2001.
- [7] A.C. Berg, T. L. Berg, J. Malik “Shape Matching and Object Recognition using Low Distortion Correspondence”, IEEE Conference on Computer Vision and Pattern Recognition (CVPR’05), 2005
- [8] Kidsroom – An Interactive Narrative Playspace. <http://vismod.media.mit.edu/vismod/demos/kidsroom/kidsroom.html>.

- [9] A.P. Pentland, C.R. Wren, F. Sparacino, A.J. Azarbayejani, T.J. Darrell, T.E. Starner, A. Kotani, C.M. Chao, M. Hlavac, K.B. Russell, "Perceptive spaces for performance and entertainment: Untethered interaction using computer vision and audition", *Applied Artificial Intelligence*, v. 11 no. 4, p. 267 - 284, 1997.
- [10] L. Emering and B. Herbelin, "Body Gesture Recognition and Action Response", *Handbook of Virtual Humans*, Wiley 2004, pp.287-302.
- [11] VIVID GROUP gesture recognition system. <http://www.vividgroup.com>.
- [12] F. Sparacino, N. Oliver, A. Pentland, "Responsive Portraits". *Proceedings of The Eighth International Symposium on Electronic Art (ISEA'97)*, 1997.
- [13] K.M.G. Cheung, S. Baker, T. Kanade, "Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'03)*, vol.1, pp.77-84, 2003.
- [14] P.F. Felzenszwalb, D.P. Huttenlocher, "Efficient matching of pictorial structures", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'00)*, vol.2, pp.66-73 vol.2, 2000.
- [15] R. Ronfard, C. Schmid, and B. Triggs, "Learning to Parse Pictures of People", *European Conference on Computer Vision (ECCV'02)*, LNCS 2353, pp.700-714, 2002.
- [16] G. Mori, X. Ren, A. Efros, and J. Malik, "Recovering human body configurations: combining segmentation and recognition", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04)*, vol.2, pp.326-333, 2004.
- [17] D. Ramanan, D. A. Forsyth, and A. Zisserman. "Strike a Pose: Tracking People by Finding Stylized Poses", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'05)*, 2005.
- [18] D. M. Gavrila and V. Philomin, "Real-time object detection for smart vehicles", *International Conference on Computer Vision (ICCV'99)*, pp.87-93, 1999.
- [19] S. Carlsson and J. Sullivan, "Action recognition by shape matching to key frames", *IEEE Computer Society Workshop on Models versus Exemplars on Computer Vision*, 2001.
- [20] G. Mori and J. Malik, "Estimating human body configurations using shape context matching", *European Conference on Computer Vision (ECCV'02)*, LNCS 2352, pp.666-680, 2002.
- [21] H. Jiang, Z.N. Li, and M.S. Drew, "Optimizing motion estimation with linear programming and detail-preserving variational method", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'04)*, 2004.
- [22] H. Jiang, M. S. Drew, and Z.N. Li, "Linear Programming Matching and Appearance-Adaptive Object Tracking", *Energy Minimization Methods in Computer Vision and Pattern Recognition (EMM-CVPR'05)*, LNCS 3757, pp. 203-219, 2005.
- [23] H. Jiang, M.S. Drew and Z.N. Li, "Successive convex matching for action detection", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'06)*, 2006.
- [24] V. Chvatal. *Linear Programming*, W.H. Freeman and Co., New York, 1983.
- [25] S. Belongie, J. Malik and J. Puzicha, "Shape matching and object recognition using shape contexts", *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol.24, pp.509-522, 2002.