光学 精密工程
Optics and Precision Engineering

文章编号　1004-924X(2009)06-1485-11

改进隐马氏模型的运动人体模型学习

苏伯超，陈　刚，车仁生
（哈尔滨工业大学 自动化测试与控制系，黑龙江 哈尔滨 150001）

摘要：基于人体模型的跟踪方法易于实现视频的运动人体跟踪，而且利用较少的视频帧数即可学习获得人体模型。本文针对给出的视频提出了学习人体模型的学习算法。利用片图模型表示未经学习的人体，改进的隐马尔可夫模型（HMM）模拟人体在视频序列各帧间的运动，并使用机器学习方法对该改进的 HMM 进行推理，获取改进 HMM 的参数，从而获得所需的人体模型。学习得到的人体模型由包含颜色信息的各人体肢体模板组成。实验显示只用 80～90 帧包含有人体运动的序列图像，便可学习得到该运动人体的人体模型。结果表明，该学习框架效果明显，可用于快速学习视频序列中的运动人体模型，且可用于学习一人或多人的人体模型。

关　键　词：机器学习；片图模型；改进的隐马氏模型
中图分类号：TP391.41　　**文献标识码：**A

Kinetic people model learning of modified HMM

SU Bo-chao，CHEN Gang，CHE Ren-sheng

(*Department of Automatic Measurement and Control*，*Harbin Institute of Technology*，*Harbin* 150001，*China*)

Abstract： The tracking method based on a people model contributes to realizing the kinetic people tracking for a given video，and it can learn the people model by using much less frames of the video. This paper proposes a learn algorithm for learning the people model in the given video. By using a tree pictorial structure model to represent the detuned generic people in the video，and a modified Hidden Markov Model(HMM) to simulate the motion of people between the two frames of the video，a machine learning method is used to the modified HMM to obtain the estimation of parameter of the modified HMM，and to capture the people model from the video. The learned model consists of different body templates covered with color information. For learning the color of the local parts of the people model by proposed algorithm，an instance-specific model has been obtained. The experiment demonstrates that the kinetic people by proposed algorithm model can be learned with sequences images of 80－90 frames involving people motion，which shows the learning method works well for learning the kinetic people model based on video，and can rapidly learn people models for one or more persons in the video.

Key words： machine learning；pictorial structure model；modified Hidden Markor Model(HMM)

Received date：2009-01-20；**Revised date：**2009-04-30.

Foundation item：Supported by the National Natural Science Foundation of China (Grant No. 50275040)

1 Introduction

Developing a kinetic people tracker, which can track people and their motion in video, is of great prospect and challenging in computer vision community. The kinetic tracker can be used for surveillance to the aim of crime prevention, for motion synthesis in games and movies, and for biomechanics.

When tracking people in frames of the given video, we need to determine where people are in the image, and which pixels belong to people bodies, so the tracking strategies have two key ideas: track by detect, track by flow[12]. The flow is pixel-pixel correspondence, is sensitive to the pose variation and illumination, and is time-wasted.

Track-by-detect means that we track people motion in video by the way of detecting people model in each frame of the video. Generally speaking, Model-based tracking is easier with better model, for people model can reduce the variation of the intra-class and increase out-of-class variability.

To build model from videos, we look to the object detection community for inspiration. There are some papers describing building object models from image collection. Cordeliade. Schmid presents a method constructing model from a set of positive and negative sample images. The positive images contain zebra, and the negative images not; this form of input is often called semi-supervised data because we label which images contain a zebra, but for a given zebra image, we do not label which image regions are zebra and which are background. The task of the learning algorithm is to "finish" the partial labeling; learn a zebra model that labels zebra image regions. Nevertheless, in our application, we do not have the negative frames from with which to compare.

R. Fergus presents a method learning the object class model from unlabeled and unsegmented cluttered scenes in a scale invariant manner. In learning, the parameters of the scale-invariant object model are estimated using expectation-maximization in a maximum-likelihood setting, and M. Weber's work is similar. EM algorithm is appropriate in our building single people model because of its inherent limitations, and can not build multiple people model if there are more people in video.

In this paper, we build a generic people model from video for the aim of tracking people. We represent people with the detuned pictorial structure in images, which consists of nine parts of people, and modified traditional HMM to model person motion between frames; We consider the model building as estimation of probability framework(modified HMM), and infer on the framework for the estimation of the model parameter, using the machine learning method[3]. Building model, in this paper, means estimation of the appearance (color) of the parts of the people model across the given video. See figure 2, it corresponds to get an instance-specific people model for the track-by-detect strategy. Thus, the processes of tracking become the process of building a good model for the people being tracked. After learning a good model from a video, we then track the people by detecting the people model in every frame.

2 Pictorial structure for representation of people

The pictorial structure model is introduced by Fichler and Elschlager[4] thirty years ago. With that model, an object is modeled by a collection of parts arranged in a deformable configuration. Each part encodes local visual properties of the objects. In this paper there are nine parts representing body segments respectively, and the deformable configuration is charactered by springlike connections between certain pairs of the parts. The model is quite general for representing the parts and the connections of the pair of parts, with great perspective and broad application in field of computer vision. Furthermore, we choose the pictorial structure for modeling people not just for learning the people model, but also for advantage of consequent model-based tracking. We can see the pictorial structure model in Fig. 1.

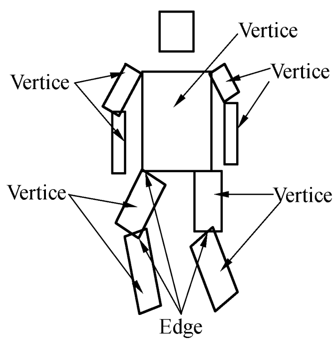
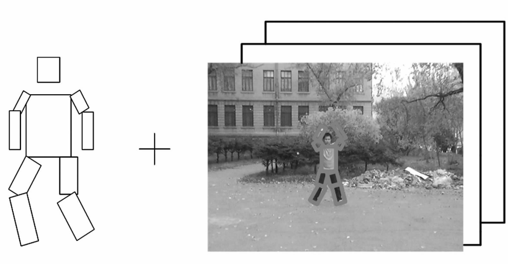


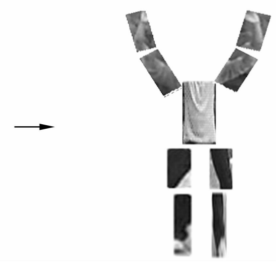
Fig. 1 Generic pictorial structure representation for people

3 Probability learning framework

We modify the hidden markov model for modeling the movement of people in video, and get an improved probability framework[5]. We then apply inference on the framework, estimating the appearance of parts[1], and finishing the learning procedure. Fig. 2 shows the learning procedure: we tune the detuned people model to the video sequence, obtain the color of the segments, and get the instance-specific people model.



(a)Detuned people model (b)Video sequence



(c)Learned people model

Fig. 2 People model learning process

3.1 Traditional HMM for tracking

By far the most common approach for tracking and learning model is to use a hidden markov model[2,10], see formula (1), where the hidden states are the poses X_t to be estimated, and the observations are images I_{m_t} of a video sequence. At present, we consider pose X_t as a vector of 2D positions. Standard Markov assumption allow us to decompose the joint probability into

$$\Pr(X_{1:T}, I_{m1:T}) = \prod_t \Pr(X_t \mid X_{t-1}) \Pr(I_{m_t} \mid X_t) , \tag{1}$$

Where we use shorthand $X_{1:T} = \{X_1, \cdots, X_T\}$. This is the basic HMM for our learning procedure[11], which can be modified for our final learning framework[9].

Learning corresponds to inference on this probability model. One searches for the Maximum a Posterior (MAP) sequence of position given an image sequence

$$\begin{aligned}\hat{X}_{1:T} = {} & \arg\max_{X_{1:T}} \Pr(X_{1:T} \mid I_{m1:T}) = \\ & \arg\max_{X_{1:T}} \Pr(X_{1:T}, I_{m1:T}) = \\ & \arg\max_{X_{1:T}} \prod_t \Pr(X_t \mid X_{t-1}) \Pr(I m_t \mid X_t) .\end{aligned} \tag{2}$$

Based on the formula (2), we develop our learning model with some modification in the position X_t. Firstly we consider building a template-matching blob tracker for a torso, see Fig. 3. The hidden variables are X_1, X_2 until X_T; the observation are I_{m_1}, I_{m_2} until I_{m_t}. The rectangle in the image is the tracking window, showing that the torso is tracked, or missed.

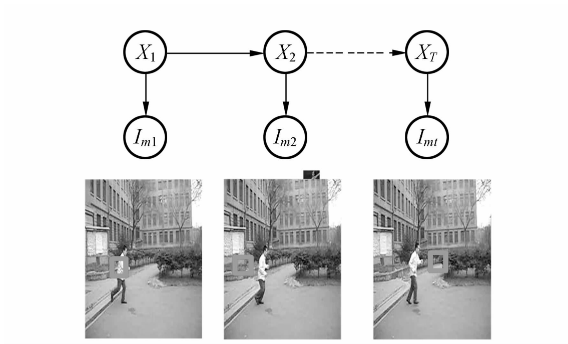


Fig. 3 Model for torso tracking

The underlying model is a HMM where the hidden state at each frame is

$$\boldsymbol{X}^i = \begin{bmatrix} P^i \\ A^i \end{bmatrix}. \tag{3}$$

Where p^i is (x^i, y^i) position of torso blob and A^i is the appearance. For the time being, we consider the A^i as a vector appearance. The observations are images from a video sequence. We assume the following probability models:

$$\Pr(X_t | X_{t-1}) \propto \mathrm{e}^{-\| X_t - X_{t-1} \|^2}, \tag{4}$$

$$\Pr(I_{m_t} | X_t) = \Pr(I_{m_t}(P_t) | A_t) \propto \mathrm{e}^{-\| I_{m_t}(P_t) - A_t \|^2}. \tag{5}$$

Where we have ignored constants for simplicity. Formula (4) is our motion model explaining the torso blob movement from one frame to another, and the torso moves in Brownian manner[5]. Recall that our object state $\boldsymbol{X}^i$ encodes both torso blob position $P_t = (x_t, y_t)$ and torso appearance A_t. formula (5) is our likelihood model, and it means that we favors pixel positions P_t at which the encompassing image patch $I_{m_t}(P_t)$ looks like the current torso blob template A_t.

If we perform inference on the probability model in Fig. 3, the resulting track tends to drift (e. g. the torso is missed in 2 latter images), because there is no constraint in the model that require the last frame A_T to be similar to the appearance from the first frame A_1. once small errors in the estimated appearance A_T accumulate, for example, the torso becomes temporarily occluded by the arm, and the track drifts to compensate. We can avoid the drifting behavior with a constant model of torso appearance C:

$$\Pr(A_t | C) \propto \mathrm{e}^{-\| A_t - C \|^2}, \tag{6}$$

Here, the torso image patch at each frame A_t is modeled as i. i. d sample from a Gaussian centered at an underlying "true" torso patch C. We can see in Fig. 4, C represents the canonical appearance; A_1, A_2 and A_T, represent the torso samples.

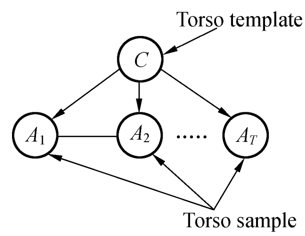


Fig. 4 Canonical appearance C

With a constant appearance model, we force our initial appearance A_1 and final appearance A_T to be similar.

3.2 Modify HMM

As we can see, traditional HMM used to track people in video has drifting behavior, if there is occlusion of the people parts, the resulting track miss people parts soon, and lose target at last. So we modify the traditional HMM for our learning process.

3.2.1 HMM with constant appearance

When we insert the constant appearance model (Fig. 4) into formula (1), we get the model of Fig. 5. From the model of Fig. 5 we can see, when tracking the torso through the video, we have P_1 and P_T represent the position of the torso in the image; I_T the appearance (color) of torso patch. The patches are similar with each other because of the constraint of the canonical appearance C.

We can see in Fig. 5, The image likelihoods are restricted by the canonical appearance C:

$$\Pr(P_t | P_{t-1}) \propto \mathrm{e}^{-\| P_t - P_{t-1} \|^2}, \tag{7}$$

$$\Pr(I_{m_t} | P_t, C) \propto \mathrm{e}^{-\| I_{m_t}(P_t) - C \|^2}, \tag{8}$$

Figure (7) expresses that the torso blob should be close between the neighbor frames,

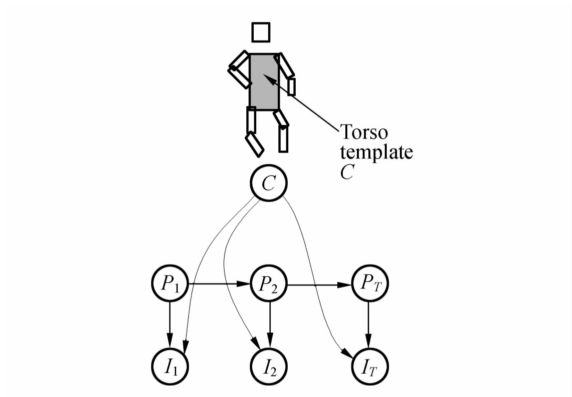


Fig 5 Constant appearance model

and Figure (8) expresses the fact that we want to select a torso blob position P_t whose encompassing image patch is close to the canonical appearance C.

If we condition on C and assume the canonical appearance is given, our model formulated by (7)(8) add a constraint appearance for standard HMM. The constant appearance model is represented by templates built a prior. In Fig. 6 we can see a HMM with constraint term, formulated by (7) (8); C: the torso template; P_T: the position of the torso; I_T: the observation.

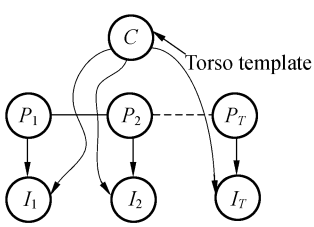


Fig. 6 HMM with constant appearance

3.2.2 The segments motion model

With people parts detector we have got the people parts patch from the video; with clustering method, we have obtained different kinds of clusters, including the cluster of people torso. See Fig. 7.

For the torso cluster, we want to find a sequence of candidates that obeys our bounded velocity motion model. By fitting an appearance model to the cluster, typically a Gaussian with mean at the cluster mean and standard deviation

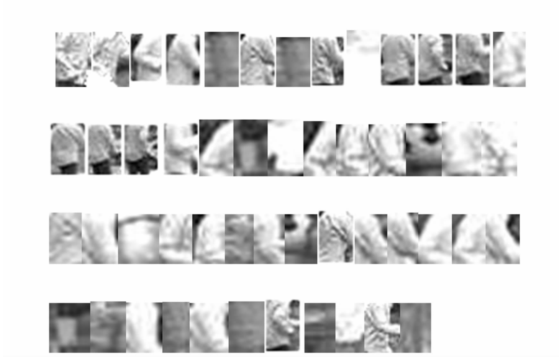


Fig. 7 Detected torso and noise

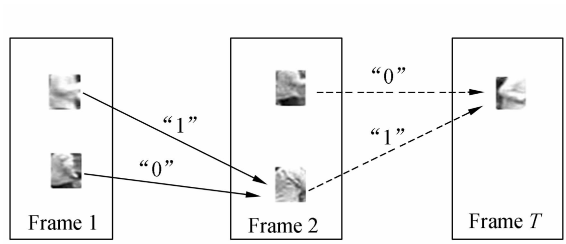


Fig. 8 Selecting torso sequence using dynamic programming

computed from the cluster, we can convert this optimization into a straightforward dynamic programming problem. we consider P_t as the position of a segment in the t^{th} frame; and We assume that these have a Markovian behavior *i. e.* $\Pr(P_t \mid P_{1:t-1}) = \Pr(P_t \mid P_{t-1})$. the score for a given candidate is its likelihood under the Gaussian appearance model we are fitting, and the temporal score are ′0′ for links violating our velocity bounds and ′1′ otherwise. We add a dummy candidate to each frame to represent a ′no match′ state with a fixed charge. by applying dynamic programming, we can obtain a sequence of segments, at most one per frame, where the segments are within a fixed velocity bound of one another and where all lie close to the cluster center in appearance. In Fig. 9, we select those torsos in the cluster of torso, utilizing the dynamic programming. Those torsos obey the motion model, as follows:

$$\psi(P_t^i, P_{t-1}^i) = \Pr(P_t^i \mid P_{t-1}^i) \propto I(\| P_t^i - P_{t-1}^i \| < d_{\max}).$$

The Fig. 9 is the torso sequence that we learn from the video data, and those torsos obey the velocity bound of the segment motion model[9].

Fig. 9 Body appearance sequence learned

3.2.3 Modifying constant-appearance HMM and learning

The discussion above is to track the single people segment in video. We use the constant appearance of people segment to solve the drift, and use the segment motion model to find segments which obey the velocity bound, those are convenient for the correct parts track. Yet we intend to model the whole people movement between frames of the video, and try to capture the appearance (color) of the people segments, so we utilize the spatial constraint of human. It means that arm must lie close to the torso, the leg must lie on the torso, and crus must lie nearby the leg. We insert the detuned people model, represented by the pictorial structure, into the Constant-Appearance HMM model formulated by (7) (8) , and get the final learning framework, as we can see in formula (13),

$$\Pr(P_{1:T}^{1:N}, I_{m1:T} \mid C^{1:N}) = \prod_{t}^{T} \prod_{i}^{N} \Pr(p_t^i \mid p_{t-1}^i) \Pr(p_t^i \mid p_t^{\pi(i)}) \Pr(I_m(P^i) \mid p_t^i, c^i) , \tag{13}$$

The Fig. 10 shows the tree configuration of people using pictorial structure in an image. The configuration is a rooted tree, showing the spatial layout of the people parts in an image.

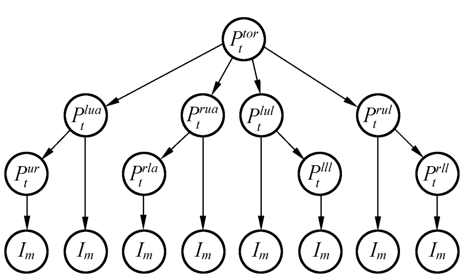


Fig. 10 Tree pictorial structure of people for learning

As a notation convention, we use superscript to denote body parts $i \in \{tor, arm, \cdots\}$ and subscripts to denote frames $t \in \{1 \cdots T\}$. the term $\pi(i)$ denotes the parent of part i, following the tree in Fig. 8. The variables p_t^i is a 3-vectos capturing the position (x, y) and orientation θ of part i at time t. The formula (13) is actually a Bayes framework which we use to explain our learning method, given the video. It means that given $C^{1:N}$, we want to obtain the maximum likelihood estimation of $\Pr(P_{1:T}^{1:N}, I_{m1:T})$. the First term in the right hand side of formula (13) is a motion model for an individual part; the last two terms are the standard geometric and local image likelihood terms in a pictorial structure.

For a fixed sequence of images $I_{m1:T}$, we can interpret the right hand side as function of p_t^i and c^i. we infer on the formula (13) for estimation of p_t^i and c^i. Effectively, we want to find an arm position P_t^{arm} such that the arm lies nearby a torso $\Pr(P_t^{\text{arm}} \mid P_t^{\text{tor}})$, the arm lies near its position in the previous frame $\Pr(P_t^{\text{arm}} \mid P_{t-1}^{\text{arm}})$, and the local image patch looks like the arm model $\Pr(I_{m_t}(P_t^{\text{arm}}) \mid P_t^{\text{arm}}, C^{\text{arm}})$.

Our image likelihood models the local image likelihood patch with a Gaussian centered at template C^i.

$$\psi_t(P_t^i, C^i) = \Pr(I_{m_t}(P_t^i) \mid P_t^i, C^i) \propto e^{-\| I_{m_t}(P_t^i) - C^i \|^2} , \tag{14}$$

For simplicity we ignore constants. The spatial kinematics of the human body is modeled with a puppet of rectangles with freely rotating revolute joints, using potential of the form

$$\psi(P_t^{\text{arm}}, P_t^{\text{tor}}) = \Pr(P_t^{\text{arm}} \mid P_t^{\text{tor}}) \propto I(D(P_t^{\text{tor}}, P_t^{\text{arm}}) < d_{\max}). \tag{15}$$

Here I the standard identity is function and $D(P_t^{\text{tor}}, P_t^{\text{arm}})$ is the distance between the hinge points for the two segments. For the upper leg segments, we add angular bounds preventing them from pointing up into the torso.

Our motion model is bounded velocity

$$\psi(P_t^i, P_{t-1}^i) = \Pr(P_t^i \mid P_t^i) \propto I(\| P_t^i - P_{t-1}^i \| < d_{\max}) . \tag{16}$$

As we can see, finding an optimal track and learning from the track given a video sequence now corresponds to finding the maximum a posteriori (MAP) estimate of C_t^i and P_t^i. The iteration of EM approach hints a useful framework to follow. Given a detuned or rough appearance model C, we can use it to obtain a rough track of the blob positions p_t. Given the rough track, we can tune the model to people in a video. Given the tuned models, we can re-track, and iterate as necessary.

3.2.4 Learning multiple parts with approximate inference

Discussion above is simply learning the appearance of torso; nevertheless, we mean to estimate the appearance (color) of all parts of the people model, and obtain the complete tuned people model. The Fig. 11 is our people model learned through the video.

Fig. 11 People model learned

We use a pictorial structure representation that models the human body as a puppet of rectangles. See Fig7. Consider tracking one or multiple people, we exploit the fact that a priori we know the geometric model $\Pr(P^i \mid P^j)$ for a human body: a torso is connected to two arms and legs. The knowledge of the geometric model should decrease the search space and time. All we need to learn is the appearance of each part $\Pr(I_m(P^i))$.

The method we take to build appearance models is a bottom-up appearance that looks for candidate body parts in each frame; we cluster the candidates to find assemblies of parts that might be people, starting from the torso[6].

Using the learned torso sequence, now we learn the appearance other parts of people model. We build the people model by searching the arm and leg nearby the torso sequence, group the assembly into the complete people model, and get the configuration of people. We accomplish this with the approximate inference[7-8], Fig. 12 shows the principal. We search the arm sequence, which obeys the velocity bound, close to the torso sequence using the known torsos and the constraint of the spatial layout of pictorial structure.

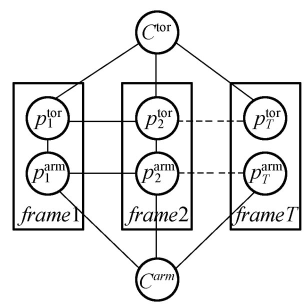


Fig. 12 Approximate inference for learning arm

With the torso sequence, as follow we learn the appearance of arm, and we explain the learning of arm template with 3 trees, the Fig. 14, Fig. 15, and Fig. 16.

We have the arm cluster with the detecting and mean-shift method, now what we do is get the arm sequence, which accords with $\psi(P_t^{arm}, P_t^{tor}) = \Pr(P_t^{arm} \mid P_t^{tor})$ and $\psi(P_t^i, P_{t-1}^i) = \Pr(P_t^i \mid P_{t-1}^i)$, that is to say, the arm sequence's locations is nearby the torso, and abide by our velocity bound. Fig. 13 is the learned torso template discussed above.

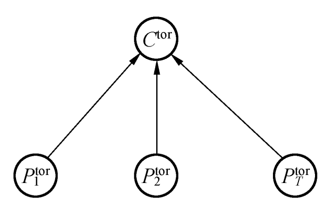


Fig. 13 Tree *a* for learning torso appearance

The inference for learning the arm template is following: we infer on tree c, as shown in Fig. 14, find the arm sequence abiding by the velocity bound, using the dynamic programming.

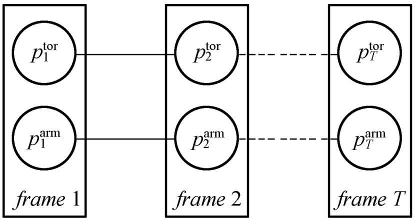


Fig. 14 Tree c for finding the arm sequence abiding by velocity bound

We get the arm sequence above, then apply inference on the tree d, see Fig. 15, and obtain the arm sequence restricted by the spatial constraint of the pictorial structure; that is, we get the arm sequence closest to the torso in each image, which is our optimal estimation of the arm sequence.

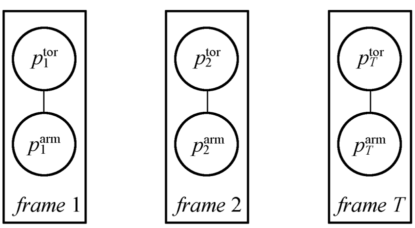


Fig. 15 Tree d for restricting arm and torso patch to those which obey kinematics constraint

We apply inference on the tree b, as shown in Fig. 16, compute the mode of the obtained arm sequence, and get the optimal estimation of C^{arm}.

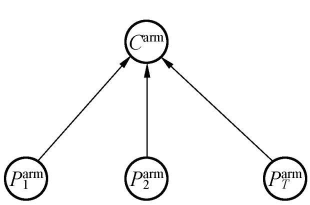


Fig. 16 Tree b for learning arm appearance

Now we get the arm template by learning. With the method above, we learn the other template by recurrence, such as the thigh, crus, each template of nine, and then assemble the people model.

4 Experiment

A given video, which longs for 10 s, consists of 120 frames; the following is the 8 frame we pick at random for demonstration, and we learn the people model from the video(Fig. 17).

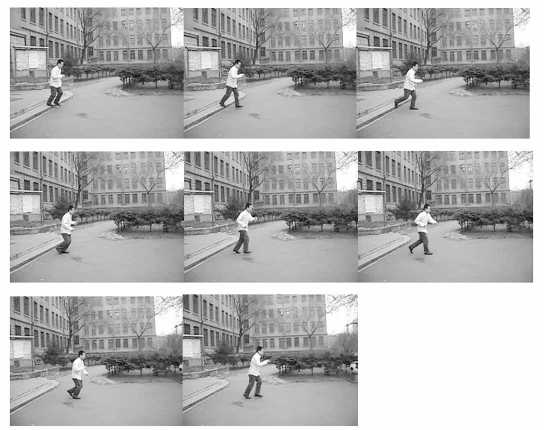


Fig. 17 Frames for learning people model (8 frames of video)

4.1 learning torso template

First of all, we learn the torso appearance, using the torso cluster with clustering method. As we can see in Fig. 18;

Fig. 18 Cluster of people torso

The torso sequence we look for must abide by the velocity bound of the segment motion model. By the dynamic programming, we obtain an accordant sequence of patches. As we see in Fig. 19, between frames, the connection "1" means that the torso patches abide by the velocity

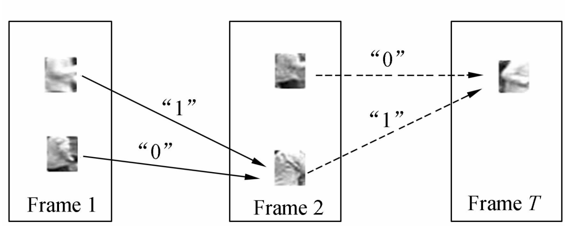


Fig. 19 Obtaining torso blobs with dynamic programming

bound, otherwise, it is"0".

We compute the mode for the later sequence, and the mode patch is the appearance of torso we learned. We can obtain the mode with the Fig. 13, tree *a*. The Fig. 20 is the torso template we learned;

Fig. 20 Learned torso template

4.2 Learning other parts appearance of people

After learning the torso template, we apply the approximate inference on tree *b c d*, (Fig. 14, Fig. 15, Fig. 16) and learn the other parts by recurrence. Inference with tree *d*, (Fig. 15) we get blobs sequence of upper arm; which is restricted by constraint of the pictorial structure, see Fig. 21.

Fig. 21 Sequence obtained by inferring on tree *d*

And inference with tree *c*, (Fig. 14), we get the sequence obey our velocity constraint, using dynamic programming; as shown in Fig. 22;

Fig. 22 Arm sequence abiding by velocity bound by inferring on tree *c*

With inference on tree *b* (Fig. 16), we obtain the template of the upper arm, with the computation of mode of sequence; as shown in Fig. 23;

Fig. 23 learned template of upper arm

For the time being, the people model only consists of two parts, see Fig. 24;

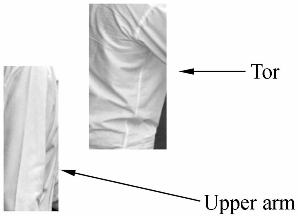


Fig. 24 People model only consisting of 2 parts

According to the Learning procedure above, we learn the other parts by recurrence, obtain the complete people model, and fulfill the learning procedure. As we can see Fig. 25;

Fig. 25 People model learned from video

4.3 Algorithm analysis

In this paper, we use modified HMM to model the people movement among the frames of the video. Using dynamic programming, We learn the templates of parts and assemble the templates into a people model. The segment templates are learned by dynamic programming, so our model-building depend on the performance of the dynamic programming.

The dynamic programming transforms exponential complexity to polynomial function. Exponential function has the complexity of $O(a^n)$, and polynomial function′s complexity is $O(n^a)$; from the viewpoint of algorithm analysis, when process has medial level inputs, polynomial function saves time compared with the exponential complexity.

As to our torso-selection procedure using dynamic programming, we select 3 torso patches possessing three maximum probabilities, and we get the complexity of $O(Tn^2)$, in which T is the observations and n is states, equaling to 3 in our procedure. In our experiment, we learn a people model with 80—90 frames, spending 6—7.5 seconds. Considering people model consisting of 9 parts, we conclude that each segment selection averagely costs 0.66 — 0.83 seconds, and the maximum time for our dynamic programming spends 0.83 seconds. Complexity $O(Tn^2) = 0.83$, and we can conclude the frames we need is inverse proportion to parts blobs needed in each frame. That is to say, when the frames increase, we can reduce the parts two or one blob, according to probability of the parts blobs.

5 Conclusions

In this paper, we propose learning algorithm for learning a people model in the given video. Considering the convenience of the computation and representation, we adopt the pictorial structure for representing generic people model, and the representation not only is in favor of model learning, but makes for the people tracking with people model.

In view of the defect of traditional HMM tracking methods, we modify the HMM by inserting canonical appearance model and bodies motion model. The modified HMM can model the people motion between frames of video. We infer on the modified HMM with machine learning and learn the parameters of the modified HMM, then finish learning the people model. The parameters are actually the bodies′ templates which are covered with different color. people model assembled with those templates is what we learned.

The experiment show that the learning algorithm work well, and just needs the frames not more than 80—90.

References:

[1] LOFFE S, FORSYTH D A. Probabilistic method for finding people[J]. *International Journal of computer vision*, 2001, 43(1): 45-68.

[2] NATH, RAVINDRA, JAIN. Using randomized search algorithms to estimate HMM learning parameters [C]. *IEEE International Advance Computing Conference*, 2009: 159-164.

[3] XIE SH R, WANG D H, LUO J. Novel stereo matching algorithm based on pair-wise DNA alignment and its implementation [J]. *Opt. Precision Eng.*, 2007, 5(1): 106-111. (in Chinese)

[4] FISCHLER M A, ELSCHLAGER R A. The representation and matching of pictorial structures[J]. *IEEE Transactions on computer*, 1973, 1(22): 67-92.

[5] TIAWONGSOMBAT P, MUN-HO J, BUM J. The most attentive person selection using HMM with multiple sources [C]. *International Conference on Control, Automation and Systems*, 2008: 1744-1748.

[6] MIKOLAJCYK K, SCHMID C, ZISSERMAN A. Human detection based on a probabilistic assembly of robust part detectors[C]. *European Conference Computer Vision*, 2004.

[7] ROVINI M, PRINCIPE F, FANUCCI L, LUISE M. Implementation of message-passing algorithms for the acquisition of spreading codes [C]. *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2008: 1441-1444.

[8] ROURKE J O, BADLER N. Model-based image analysis of human motion using constraint propagation [J]. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 1980(2): 522-546.

[9] RAMANAN D, FORSYTH D A. Finding and tracking people from the bottom up [J]. *Computer vision and pattern recognition*, 2003, 2: 467-474.

[10] LI G, WANG M J, LIN L, ZENG R L. Lip contour description based on orthogonal transform in visual driven speech synthesis system [J]. *Opt. Precision Eng.*, 2007, 15(7): 1117-1123. (in Chinese)

[11] WANG M J, LI G, LIN L, ZENG R L. Feature analysis of lip's image sequences based on weighted combination [J]. *Opt. Precision Eng.*, 2008, 16(3): 511-517.

[12] DAVID A F, OKAN A, LESLIE I. *Computational Studies of Human Motion: Tracking and Motion Synthesis* [M]. 2006: 83-103.

Authors' biographies:

SU Bo-Chao (1976－), male, Ph. D. candidate, received his M. S. Degree from Harbin Institute of Technology in 2004. His research focuses on computer vision. **E-mail**: subochao97@126. com

CHEN Gang (1971－), Male, Ph. D., associate professor of the Department of Automatic Measurement and Control, Harbin Institute of Technology, he received his M. S. and Ph. D. degrees in Harbin Institute of Technology in 1999, and in 2005 respectively. His research focuses on machine vision. **E-mail**: chengang@hit. edu. cn

CHE Ren-sheng (1939－), male, Ph. D. tutor, professor of the Department of Automatic Measurement and Control, Harbin Institute of Technology, his researches focus on bionics metrology, intelligence metrology, impersonate coordinate measuring technology, 3D dynamic measuring technology, and computer vision. **E-mail**: crs@hit. edu. cn

●**下期预告**

莫尔条纹信号相位误差补偿

吕孟军[1]，郭　琪[1,2]，吕印晓[1]

(1. 空军第一航空学院，河南 信阳 464000；

2. 南京航空航天大学 能源与动力学院，江苏 南京 210016)

为减小莫尔条纹信号不正交时的正切法细分误差，提出了一种可对任意相位滞后误差进行实时补偿的新算法。在分析出相位不正交对细分精度的影响后，通过对信号过零点时的准确采样，计算出余弦信号相位滞后的角度值，进而确定实际的相位计算公式；根据存在相位滞后信号的极性和幅值信息，将完整的短周期信号进行相位分段补偿，并分析了影响算法实现的各个因素。仿真实验表明，本算法可实现相位滞后误差的实时补偿，较好地改善信号相位不正交对细分精度的影响，使细分误差仅为未补偿误差的10%，能够极大提高莫尔条纹信号细分精度和位移检测精度。