Recognition of human activities using SVM multi-class classifier

Huimin Qian *, Yaobin Mao, Wenbo Xiang, Zhiquan Wang

School of Automation, Nanjing University of Sci. & Tech., Nanjing 210094, PR China

A R T I C L E   I N F O

Article history:
Received 2 December 2008
Received in revised form 12 September 2009
Available online 19 September 2009

Communicated by N. Pears

Keywords:
Human activity recognition
Background subtraction
CCMEI
Support vector machine
Decision tree classifier

A B S T R A C T

Even great efforts have been made for decades, the recognition of human activities is still an immature technology that attracted plenty of people in computer vision. In this paper, a system framework is presented to recognize multiple kinds of activities from videos by an SVM multi-class classifier with a binary tree architecture. The framework is composed of three functionally cascaded modules: (a) detecting and locating people by non-parameter background subtraction approach, (b) extracting various of features such as local ones from the minimum bounding boxes of human blobs in each frames and a newly defined global one, contour coding of the motion energy image (CCMEI), and (c) recognizing activities of people by SVM multi-class classifier whose structure is determined by a clustering process. The thought of hierarchical classification is introduced and multiple SVMs are aggregated to accomplish the recognition of actions. Each SVM in the multi-class classifier is trained separately to achieve its best classification performance by choosing proper features before they are aggregated. Experimental results both on a home-brewed activity data set and the public Schüldt’s data set show the perfect identification performance and high robustness of the system.

© 2009 Elsevier B.V. All rights reserved.

1. Introduction

Human motion analysis in computer vision involves object detection, tracking and recognition of human activities. Among which, activity recognition has a wide range of promising applications in security surveillance, human machine interaction, entry/exit control, sports and video annotation, etc. A large amount of work has been done on the active topic over the past decades, as the reviews (Aggarwal et al., 1999; Wang et al., 2003; Gavrila, 1999; Moeslund et al., 2006; Poppe, 2007) have summarized, nevertheless it is still an open and challenging problem.

As in many vision-related problems, feature extraction is an element operation for the recognition of human activities. Two methods, namely, model-based approach and model-free approach, are generally used (Aggarwal et al., 1999). In many situations, the model-based approach is not suited to practical applications due to its complexity which suffers from the generally used 2D shape models or 3D volumetric human models. While in the model-free approach, human movement is usually represented by low-level visual features from the region of interest. Shape, kinematics information such as body contour, centroid trajectory, optical flows and motion speed are usually used to recognize gestures, gait and certain behaviors (Min and Kasturi, 2004; Psarrou et al., 2002; Veeraraghavan et al., 2005). However, it is not an easy task to stably extract the aforementioned features from the real-scene images.

For example, due to the occlusion or the color resemblance between foreground and background, some parts of a human blob may be undetected by generally used object detection methods. Therefore, some features cannot be obtained stably, such as motion energy image (MEI) and motion history image (MHI), which are introduced by Bobick et al. (2001) to characterize human motion information in image sequences. MEI is a binary image that encodes motion occurrence, while MHI is a multiple-value image that encodes the recency of motion occurrence. MHI and its extensions (Weinland et al., 2006; Xiang et al., 2002; Meng and Pears, 2009) are commonly used in motion analysis. In order to overcome the segmentation errors, a new feature named contour coding of the motion energy image (CCMEI) is proposed to describe the actions in this paper. The feature is not only steady but invariant to translation and scale change.

In machine learning, there are two main approaches to perform classification (Rubinstein and Hastie, 1997): generative learning or discriminative learning.

It has been shown that discriminative classifiers often achieve better performance than generative classifiers (Ng and Jordan, 2002) in supervised learning. For example, support vector machines (SVMs) (Lee and Xu, 2004) directly maximize the margin of a linear separator between two sets of points in the vector space. Since the model is linear and simple, the maximum margin criterion is more appropriate than maximum likelihood or other generative model criteria. In the image-based digit recognition, SVMs have produced state of the art classification performance (Vapnik, 1995, 1998). In text classification (Rennie and Rifkin,
support vector machines surpassed the popular naïve Bayes and generative text models. In computer vision, person detection and recognition (Nakajima et al., 2000) have been dominated by SVM frameworks which surpass maximum likelihood generative models. Thus, in activity classification with labelled training samples in this paper, support vector machines would be superior to the generative classifiers. Besides, the thought of hierarchical classification is introduced to achieve the multi-class classification.

In this paper, a smart surveillance system is proposed, whose framework is described as follows:

- Detecting and locating people by background subtraction approach.
- Extracting various of features including shape information and motion information.
- Recognizing activities of people by SVM multi-class classifier with binary tree architecture.

Intelligent home-care systems which can especially detect the fall action are much-needed in the aging society. In this paper, an intelligent home-care surveillance system is constructed. Owing to the lack of corresponding public data set about the daily actions, especially with the fall action, a home-brewed activity data set is set up, which is composed of six kinds of activities, including walking, jogging, stand-to-sit, stand-to-squat, fall, and in-place actions like standing, sitting and squat. The data set has been used in our previous work (Qian et al., 2008).

In the surveillance system, a stationary camera mounted on the wall is employed to collect the surveillance videos. Commonly used background subtraction is adopted to obtain moving human blobs, and the background pixel is estimated by non-parameter method. Both shape information and motion information including the speed, the height, the aspect ratio and the newly defined feature, CCMEI are used to depict the activity. In the recognition phase, the SVM decision tree is used to learning the boundaries between activity classes. Differ from all classes are considered simultaneously in previous approaches (Wang et al., 2007; Ke et al., 2005), we perform activity classification in a hierarchical manner. Each node on the decision tree is an SVM binary classifier, and all the SVMs are aggregated by a binary tree to form a multi-class classifier named support vector machines with the binary tree architecture (SVM-BTA) (Cheong et al., 2004). The proposed surveillance system can not only label the activity, but detect the abnormal activity fall and issue an alarm for the abnormality to implement the intelligent home-care.

In addition, another system is constructed to recognize the activities in the public database (Schüldt et al., 2004). Experimental results of both systems illustrate the efficiency and the effectiveness of the proposed method.

The rest of the paper is organized as follows. Section 2 deals with related works. Section 3 describes the extraction of the human blobs by background subtraction and the formation of the features. Section 4 introduces the structure design of the multi-class support vector machine classifier with the binary tree architecture (SVM-BTA). Then, Section 5 reports the training of the classifier as well as the recognition results on a home-brewed data set and a public database. Finally, Section 6 concludes the paper.

2. Related work

Numerous efforts have been made to recognize human activities by wearable-sensors-based systems (Ermes et al., 2008; Maurer et al., 2006) or vision-based systems (Zaidenberg et al., 2006; Niu and Abdel-Mottaleb, 2005). The effectiveness of the sensors-based methods depends on the cooperativity of the person being observed. On the contrary, in vision-based systems, people no longer need to wear anything and all the recognition work is done by cameras and computers. Actually, the vision-based system can give more information of a person’s behaviors from surveillance video than the sensors-based system can do.

Each of the referred surveys of human motion analysis in computer vision (Aggarwal et al., 1999; Wang et al., 2003; Gavrila, 1999; Moeslund et al., 2006; Poppe, 2007) has its specific focus and taxonomy. Thereinto, Aggarwal et al. (1999), Wang et al. (2003) and Poppe (2007) all divide the methods of human motion analysis into two classes: model-based and model-free approaches. The former employs an a prior shape model, in which 2D and 3D explicit models are used to describe the human body, while the latter only needs to implicitly model the relation between image observation and pose variations. The following are some examples of the model-based methods. Feng and Perona (2002) recognize human actions by movelet codewords which collect the shape, motion and occlusions of the 10 parts of a human body model. Ben-Arie et al. (2002) recognize 8 activities, like jumping, knelling, picking up or putting down an object, running, sitting down, standing up and walking, by model-based method. The activity is described by a set of pose and velocity vectors of the major body parts (hands, legs, and torso) and indexed by a set of multidimensional hash tables. The feature vectors employed in the model-based approaches are mostly from the body models which seem difficult to construct.

In the model-free approaches, features extracted from image sequences are usually employed as action descriptors. Generally used descriptors are derived from silhouette and contour, optical flow, trajectory, interest points or other features, among which silhouette and contour are mostly adopted.

A spatiotemporal volume (STV) generating from a sequence of 2D contours of human blob is presented by Yilmaz and Shah (2005a). The action descriptors computed by analyzing the differential geometric properties of STV are used to perform action recognition.

Kellokumpu et al. (2005) propose a system to recognize 15 human activities only with the silhouette features, and it adopts the SVM classifier for posture classification and discrete hidden Markov models (dHMMs) for activity recognition.

Efros et al. (2003) propose a motion descriptor based on blurred optical flow measurements plus normalized correlation to recognize ballet, tennis, and football actions at a distance.

Schüldt et al. (2004) achieve action recognition using local measurements in terms of spatiotemporal interest points with SVM classifier. Dollár et al. (2005) also present a spatiotemporal interest points detector, as Schüldt et al. (2004) do, and a number of cuboid descriptors to construct behavior descriptors.

Yilmaz and Shah (2005b) make use of 13 landmark’s trajectories together with dynamic epipolar geometry matching to recognize human actions recorded by moving cameras.

Yacoob and Black (1999) obtain principle curves of actions in terms of trajectories of five body parts and recognize 4 actions like walk, marching, line-walk and kicking while walk.

Madabhushi and Aggarwal (1999) recognize actions in the frontal or lateral view by tracking the movement of the head of the subject, however, the amount of the actions considered in the paper is few.

We note that the features in some of these methods are not stable enough (Efros et al., 2003), and cannot be extracted exactly for real applications (Yilmaz and Shah, 2005b; Yacoob and Black, 1999; Madabhushi and Aggarwal, 1999), which would make the recognition performance decrease. Some other features are stable enough (Yilmaz and Shah, 2005a; Dollár et al., 2005; Schüldt et al., 2004), but the accuracy of the recognition method is not discouraging. In order to achieve better accuracy, in this paper, a
new stable and easily extracted feature CCMEI and the thought of hierarchical classification are both introduced to recognize human actions in real scenes.

3. Feature extraction of activities

3.1. Background modelling and subtraction

The first step in our proposed system is to segment motion targets in a surveillance video shot by a camera mounted on the wall. Detecting moving blobs provides a focus of attention for later processes like activity recognition, where only those changing pixels are subject to consideration. However, changes of illumination, shadow and repetitive motion from clutter make the segmentation unreliable. Several motion detection approaches such as background subtraction, temporal difference, and optical flow are generally used, where the background subtraction is particularly popular especially under the situations with a relatively static camera. It attempts to detect moving regions in an image by subtracting a reference background image from the current image pixel-wisely. In our case, a non-parameter background model developed by Elgammal et al. (2000) is adopted, where the probability density function at each pixel is worked out from multiple samples by kernel density estimation technique.

Suppose that \(x_1, x_2, \ldots, x_N\) are \(N\) temporally sampled features in a pixel position, the observation at time \(t\) is \(x_t\), and its probability density function can be estimated using kernel density as:

\[
p(x_t) = \sum_{i=1}^{N} \alpha_i K_h(x_t - x_i)
\]

where \(K_h\) is a kernel function with window length \(h\), \(\alpha_i\) is the normalized coefficient, and usually \(\alpha_i = 1/N\).

In (Elgammal et al., 2000), kernel function \(K_h\) is chosen as normal distribution \(N(0, \Sigma)\), where \(\Sigma\) is kernel bandwidth. Supposing the kernel bandwidth of three color components is independent one another, and the bandwidth of the \(j\)th component is \(\sigma_j^2\), then \(\Sigma = \text{diag}(\sigma_1^2, \sigma_2^2, \sigma_3^2)\), where \(\text{diag}(\cdot)\) denotes a diagonal matrix. The probability density function (pdf) of \(x_t\) can be written as:

\[
p(x_t) = \frac{1}{N} \sum_{i=1}^{N} \prod_{j=1}^{d} \frac{1}{\sqrt{2\pi\sigma_j^2}} e^{-\frac{(x_t - x_i_j)^2}{2\sigma_j^2}}
\]

where \(d\) is the number of color components. The kernel bandwidth \(\sigma\) can be estimated by calculating the median absolute deviation over samples. A pixel is considered as the foreground pixel if \(p(x_t) < t_h\), where the threshold \(t_h\) is a global threshold. In the model-updating phase, a long-term model and blind update are used. In order to eliminate illumination influence and noise, shadow detection and removal, morphologic processing and connectivity component analysis are employed. The process of extracting the binary human blob image is shown in Fig. 1, where the parameters for the indoor circumstance are set as follows: the number of samples \(N = 10\), update interval \(\tau = 100\), and the threshold \(t_h = e^{-10}\).

3.2. Feature extraction

The extracted features can be classified into two categories: the local feature extracted from each frame, and the global feature found through the whole activity sequence.

3.2.1. Local feature

The local features are all extracted from the minimum bounding boxes, as shown in Fig. 2, of human blobs detected by the method depicted in Section 3.1. There are three merits to select the minimum bounding box as a feature source: first, it is generally easier to find a box to bound a blob compared with other approaches that need complicated models or cumbersome computation like silhouette extraction; second, dynamic information extracted from the bounding boxes is easy to process; lastly, the most important reason is that the measurements extracted from the bounding box are more stable than those directly extracted from the detected binary images. The extracted local features include shape features and motion features.

3.2.1.1. Shape features. Two kinds of shape features are extracted from the minimum bounding box. One is the unitary height \(h(t) = H(t)/H_{\text{max}}\), where \(H(t)\) is the height of the bounding box at instant \(t\), \(H_{\text{max}}\) is the maximum value of \(H(t)\) among the entire activity sequence. The other is the ratio of height against width \(R(t) = H(t)/W(t)\), where \(W(t)\) is the width of the bounding box at instant \(t\).

3.2.1.2. Motion features. The motion features are based on the centroid of human blob. Suppose that \(f(x, y)\) is an image, the geometric moment of \(p + q\) rank can be defined as

\[
m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x, y)
\]

![Fig. 2. The minimum bounding box of a human blob.](image_url)

![Fig. 1. Extraction of human blob.](image_url)
where $M$ and $N$ are the width and height of the image, respectively. Then, the centroid coordinates are defined as

$$
(x_0, y_0) = \left( \frac{m_{10}}{m_{00}}, \frac{m_{01}}{m_{00}} \right).
$$

Thus, the centroid coordinates $(x_0(t), y_0(t))$ of human blob at time $t$ can be calculated. The motion features involve the instantaneous speed $v(t)$, the speed along $x$-axis $v_x(t)$ and the speed along $y$-axis $v_y(t)$ of the centroid. They are defined, respectively as

$$
v_x(t) = x_0(t) - x_0(t - 1)
$$

$$
v_y(t) = y_0(t) - y_0(t - 1)
$$

$$
v(t) = (v_x(t), v_y(t))
$$

And the amplitude and angle of $v(t)$ can be calculated as

$$
|v(t)| = \sqrt{v_x^2(t) + v_y^2(t)} \quad \text{and} \quad \theta(t) = \arctan(v_y(t)/v_x(t)),
$$

respectively.

3.2.2. Global feature

Bobick et al. (2001) had ever introduced motion energy image (MEI) and motion history image (MHI) as temporal templates to characterize an action. The MHI is a function of the action duration and dependent on the entire extraction of moving targets in each frames, which is difficult in real scenes. The extensions of MHI referred in Section 1 also have the same limitations. The binary version of the MHI is MEI, in which hollows exist when parts of human blob are undetected. However, the contour of MEI is correspondingly stable. Thus, a new type of feature named contour coding of the motion energy image (CCMEI) from the contour of the MEI is presented. In order to make the feature invariant to scale and translation transformation, the scale standardization and square-to-circular coordinates transformation are performed.

Let $I(x, y, t)$ be an activity sequence, $D(x, y, t)$ be the corresponding binary image sequence indicating regions of motion. In this paper, $D$ is generated by background subtraction with some postprocessing. The MEI is defined (Bobick et al., 2001) as

$$
E_t(x, y, t) = \sum_{i=0}^{t-1} D(x, y, t - i)
$$

where $t$ is the temporal extent of an action. Fig. 3a and b show the MEI of walk and stand-to-sit activities, respectively. The MEI of walk action shown here has hollows due to segmentation error, which affects the contour little. Before the contour are picked up, the MEI is adjusted to the same height to eliminate the effect of the imaging distances. The adjusted contours of the two samples are also shown in Fig. 3.

The new feature, CCMEI calculated from the contour of the adjusted MEI is extracted by square-to-circular transformation and contour coding.

3.2.2.1. Square-to-circular transformation. In order to extract the contour feature, the plane coordinates are changed into the polar coordinates by a square-to-circular transformation (Mukundan and Ramakrishnan, 1995). The pixels on the image are considered as points in the concentric squares and mapped onto concentric circles. See in Fig. 4. Suppose the image with $N \times N$ pixels has a coordinate system $(X, Y)$ with origin at the center of the image, then $-N/2 \leq X, Y \leq N/2$. The interm variables $\rho, \sigma$ are used to calculate the corresponding pixel coordinates in the polar coordinates. The variable $\rho$ denotes the radius of the concentric circle and $\sigma$ the position index of the pixel on the circle. Both $\rho, \sigma$ are integrals and they are defined as

$$
\rho = \begin{cases} 
|X| - |Y| & \text{if } \rho \geq |Y| \\
|Y| & \text{others.} 
\end{cases}
$$

$$
\sigma = \begin{cases} 
2(\rho - X) \frac{\pi}{\rho} + \frac{\pi}{2} & |X| \geq |Y| \\
2Y - \frac{\pi}{2} & \text{others.} 
\end{cases}
$$

The radius $\rho$ takes values from $1$ to $N/2$ along the radial direction and the index $\sigma$ is from $-4\rho$ to $4\rho$ along the circumference, whose transformation directions are illustrated in Fig. 4. Thus, the polar coordinates $(r, \theta)$ are

$$
r = 2\rho/N
$$

$$
\theta = \pi\sigma/(4\rho)
$$

Define the centroid of the MEI as $(x_c, y_c)$, around which a minimum bounding square of the MEI is extracted. The size of the square is $N \times N$. And $N$ is the maximum one of the height and width of the MEI. Choose the centroid $(x_c, y_c)$ as the origin, and calculate the polar coordinates $(r, \theta)$ by the Eqs. (7) and (8).

3.2.2.2. Contour coding. Divide the scope of $\theta$, $[0, 2\pi]$ into 180 intervals, and combine the corresponding values $r$ of each intervals into a vector, which is just the new feature CCMEI.

4. SVM multi-class classifier with binary tree architecture

In supervised learning, it has been shown that discriminative classifiers often achieve better performance than generative classifiers (Ng and Jordan, 2002). Support vector machine is a successful representative of discriminative classifiers. Traditional classifiers such as Naive Bayes, BP network and KNN may not work well in the case of limited samples due to the prouness of over-fitting, while SVM does. SVM has good generalization ability as it is based on the principle of the structural risk minimization in statistic learning theory, while activity recognition is exactly a classification problem with limited samples, where SVM classifier just finds its usage.

4.1. SVM classifier

The theory of the support vector machine (SVM) is advanced by Vapnik (1995, 1998). It has yielded excellent results in various two class classification problems such as handwritten digit recognition, face detection in images, and text categorizations in recent years. Due to the perfect learning performance, SVM has become a research hotspot in machine learning.

SVM classifier deals with two-category classification problems. Given a training sample set $\{(x_i, y_i), i = 1, \ldots, n; x_i \in \mathbb{R}^d, y_i \in \{-1, +1\}\}$, where $x_i$ is the feature vector, $y_i$ is the label, SVM is developed for finding the optimal classification plane in the case of linear separability. The aim of SVM classifier is to maximize the margin between two categories besides distinguishing them.

Under the case of linear separability, the optimal hyperplane can be constructed by solving an optimization problem:

$$
\min \theta \Phi(w) = \frac{1}{2} (w \cdot w)
$$

s.t. $y_i ((w \cdot x_i) + b) \geq 1, \quad i = 1, \ldots, n$

whose dual problem is:

$$
\max W(x) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j)
$$

s.t. $\sum_{i=1}^{n} \alpha_i y_i = 0, \quad \alpha_i \geq 0, \quad i = 1, \ldots, n$

If $x^*$ is a solution of Eq. (10), then $w = \sum_{i=1}^{n} \alpha_i y_i x_i$. Choose an $x_i \neq 0$, and the corresponding solution $b$ is computed from the
equation $a_i (y_i (w \cdot x_i + b) - 1) = 0$. Then the label of an unknown sample $x$ can be decided through $\text{sgn}[w \cdot x + b]$.  

In practical applications, the linearly separable condition cannot always be satisfied in most of the case. Therefore, a relaxation variable $\xi_i \geq 0$ and a mapping $\phi(x)$ are introduced to get a nonlinear support vector machine. The optimal hyperplane can also be constructed by calculating the following optimization problem:

$$
\min \Phi(w) = \frac{1}{2} (w \cdot w) + C \sum_{i=1}^{n} \xi_i \\
\text{s.t.} \quad y_i ((w \cdot \phi(x_i)) + b) \geq 1, \quad i = 1, \ldots, n
$$

where $C$ is a penalty factor. Its dual problem is:

$$
\max L(a) = \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i,j=1}^{n} a_i a_j y_i y_j K(x_i, x_j) \\
\text{s.t.} \quad \sum_{i=1}^{n} a_i y_i = 0, \quad 0 \leq a_i \leq C, \quad i = 1, \ldots, n
$$

where $K(x_i, x_j)$ is a kernel function satisfying the Mercer condition (Vapnik, 1998). There are several commonly used kernel functions, such as linear kernel, RBF kernel, and polynomial kernel (Vapnik, 1998).

Fig. 3. Activities and the corresponding MEI.

Fig. 4. Schematic illustration of square-to-circular transformation.
4.2. Multi-class SVM classifier

Although support vector machine classifier is originally developed for two-category problems, it can be extended to multi-class classification by two popular approaches (Hsu and Lin, 2002). One is to combine a number of two-category classification SVMs in a certain manner to form a multi-class classifier, while the other is to directly solve a multi-class classification function with the training samples (Weston and Watkins, 1999). The decision-making function of the latter is difficult to work out and the training and testing processes are also time-consuming. Therefore, the former method is more practical and several algorithms were derived including the one-against-rest method, the one-against-one method, DAG-SVM, ECOC-SVM, and SVM-BTA.

4.2.1. One-against-rest

The one-against-rest method is probably the earliest implementation for SVM multi-class classifier (Vapnik, 1998). It needs \( N \) SVM classifiers for an \( N \)-class classification problem. All of the samples are considered when training each SVM, and the samples of a certain class have the positive label while the remainders have the negative label. Therefore, the training process for all the \( N \) classifiers consumes long time. Besides, the great quantitative discrepancy between the negative and the positive samples in the classification problem with many classes would degrade the classification performance. Another severe shortcoming of this method is that there are unclassified samples.

4.2.2. One-against-one

For \( N \)-class classification problem the one-against-one method (Kreßel, 1999) needs \( \frac{N(N-1)}{2} \) classifiers, each of which is trained on samples from the two corresponding classes. After all the classifiers are trained, a voting strategy is used for test. The unlabelled sample is assigned to the class with the largest vote. Compared with the one-against-rest method, the classification accuracy is improved, however, there are still unclassified samples.

4.2.3. DAG-SVM

The directed acyclic graph SVM (DAG-SVM) (Platt et al., 2000) is based on the decision directed acyclic graph (DDAG). In the same way as the one-against-one method, \( \frac{N(N-1)}{2} \) SVM classifiers are constructed independently in the training process for \( N \)-class classification problem, while in the test process the binary SVMs are combined according to the bidirectional DDAG. Compared with the one-against-one method, the advantages of DAG-SVM include quicker classification speed under the alike accuracy and lack of unclassified samples. Whereas, the construction of DDAG consumes much time and the structure of DDAG is precarious.

4.2.4. ECOC-SVM

The error correcting codes SVM (Dietterich and Bakiri, 1995) presented by Dietterich and Bakiri considers the class as correcting codes. The class label of each sample is a coding, each digit of which is designed by an SVM. A sample is classified into the class with the less Hamming distance from the pending sample. Whereas, the design of the code table and the choice of the best array are difficult for practical problems.

4.3. SVM-BTA

SVM-BTA (Cheong et al., 2004; Takahashi and Abe, 2002) is a multi-class classifier with a hierarchical binary tree architecture, of which each node makes a binary decision by an SVM. Optional characteristics and decision rules can be made for each node separately. It only needs \( N-1 \) SVMs for \( N \)-class classification problem, therefore, the training consumption is low. In the test process, parts of SVMs instead of all the SVMs are subjected to operation, therefore the calculation payload is low. Furthermore, similar to DAG-SVM, there are no unclassified samples by this method. The architecture design of an SVM-BTA is most important as the misclassification in the upper level would propagate along the tree. Thus, the classification principle of the SVM-BTA is to make the classes easy to differentiate distinguished first. Based on the assumption that the larger the distance between the classes, the easier to differentiate, the clustering method is used, in this paper, to devise the structure of the SVM-BTA. Here, the results of the \( K \)-means clustering are chosen as the splitting criterion in building the decision tree.

4.3.1. The structure of the SVM-BTA

Take experiments on the home-brewed data set as examples to illustrate the process of structure design of the SVM-BTA. There are six types of activities, and the problem of recognizing the six actions is broken into five distinct binary classification sub-problems. According to the classification principle aforementioned, the most segregative activities should be distinguished first. Thus the results of the \( K \)-means clustering are used as the splitting criterion in building the SVM-BTA, that is in the nodes with the pending activities the \( K \)-means clustering is firstly employed to divide the activities into two classes. Since the clustering results are affected by the features and the distance measurements, the cluster veracity and the cluster silhouette value are employed to choose the best clustering result. The cluster silhouette value of a sample is the ratio of within-class scatter against between-class scatter, whose range is from \(-1\) to \(+1\).

The feature vectors employed in this paper include CMEI, \( h = (h(t), t = 1, \ldots, \tau) \), \( R = (R(t) - R_0, t = 1, \ldots, \tau) \), \( V = (V(t), t = 1, \ldots, \tau) \), and \( V = (V(t), t = 1, \ldots, \tau) \), where \( \tau \) is the duration of the activity, \( R_0 \) is the mean value of \( R(t), t = 1, \ldots, \tau \), as detailed in Section 3.2.2. Four distance measurements including sqEuclidean, cityblock, cosine, and correlation are used in the \( K \)-means clustering with \( K = 2 \).

The pending activity samples consisting of six kinds of actions are firstly clustered into two classes by \( K \)-means clustering. And the clustering result is considered as the splitting criterion of the root node of the SVM-BTA. Five types of feature combinations including CMEI, \( h, R \), \{CMEI, \( h \), \} (CMEI, \( R \) ) and four distance measurements aforementioned are used in the clustering. The clustering results of 150 samples under the different combinations of features and distance measurements are listed in Table 1. The instance combination of the correlation distance measurement and the feature \( h \) failed for the small relative standard deviation on some samples. In the table, three kinds of results emerge frequently, such as \{\( C_{11}, C_{12} \), \{\( C_{21}, C_{22} \), \} \{\( C_{31}, C_{32} \), \} where \( C_{11} \) denotes \{walk, jogging, fall\}, \( C_{12} \) denotes \{in-place actions, stand-to-sit, stand-to-squat\}, \( C_{21} \) denotes \{walk, jogging\}, \( C_{22} \) denotes \{in-place actions, fall, stand-to-sit, stand-to-squat\}, \( C_{31} \) denotes \{walk, jogging, in-place actions\}, and \( C_{32} \) denotes \{fall, stand-to-sit, stand-to-squat\}. The case \{\( C_{21}, C_{22} \)\} is out of court because of the big error rate. Furthermore, the cluster error rate is zero if and only if the case \{\( C_{11}, C_{12} \)\} comes forth, which points out that these two cluster sets can be partitioned effortlessly.

Besides the clustering veracity, the cluster silhouette value, which is defined as the ratio of within-class scatter against between-class scatter, is introduced to measure the clustering performance. The closer to \(+1\) the silhouette value is, the better the cluster result is. The silhouette curves of the cluster cases \{\( C_{11}, C_{12} \)\} and \{\( C_{31}, C_{32} \)\} are shown in Fig. 5. In Fig. 5a and b, not only the clustering results of \{\( C_{11}, C_{12} \)\} are invariant under different combinations of features and distance measurements, but the silhouette values of the samples are all greater than zero and even
most of them are more than 0.5. Whereas, the clustering results of \(C_{31}, C_{32}\) are variant under different combinations of features and distance measurements, and the silhouette values of some samples are less than zero which means the samples are misclassified. Therefore, the clustering result of \(C_{11}, C_{12}\) is more encouraging.

According to the clustering result, the two child nodes of the root node of the SVM-BTA are the activities sets \{walk, jogging, fall\} and \{in-place actions, stand-to-sit, stand-to-squat\}. The remained parts of the SVM-BTA are determined by the same way. Finally, the architecture of the SVM-BTA on home-brewed data set is listed in Fig. 6.

**Table 1**

<table>
<thead>
<tr>
<th>Distance</th>
<th>Features</th>
<th>Cluster 1</th>
<th>Cluster 2</th>
<th>ER (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>sqEuclidean</td>
<td>(C_{11})</td>
<td>(C_{12})</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Cityblock</td>
<td>(C_{11})</td>
<td>(C_{12})</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>Cosine</td>
<td>CCMEI</td>
<td>(C_{11})</td>
<td>(C_{12})</td>
<td>0</td>
</tr>
<tr>
<td>Correlation</td>
<td>(C_{11})</td>
<td>(C_{12})</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>sqEuclidean</td>
<td>(C_{21})</td>
<td>(C_{22})</td>
<td>11.33</td>
<td></td>
</tr>
<tr>
<td>Cityblock</td>
<td>(C_{21})</td>
<td>(C_{22})</td>
<td>11.33</td>
<td></td>
</tr>
<tr>
<td>Cosine</td>
<td>CCMEI</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>4</td>
</tr>
<tr>
<td>Correlation</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>sqEuclidean</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Cityblock</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Cosine</td>
<td>(R)</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>5.33</td>
</tr>
<tr>
<td>Correlation</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>2.67</td>
<td></td>
</tr>
<tr>
<td>sqEuclidean</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>11.33</td>
<td></td>
</tr>
<tr>
<td>Cityblock</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>11.33</td>
<td></td>
</tr>
<tr>
<td>Cosine</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>Correlation</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>sqEuclidean</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>7.33</td>
<td></td>
</tr>
<tr>
<td>Cityblock</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>7.33</td>
<td></td>
</tr>
<tr>
<td>Cosine</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>24</td>
<td></td>
</tr>
<tr>
<td>Correlation</td>
<td>(C_{31})</td>
<td>(C_{32})</td>
<td>20</td>
<td></td>
</tr>
</tbody>
</table>

Fig. 5. The clustering silhouette curves of the two cases \(C_{11}, C_{12}\), \(C_{31}, C_{32}\) under different features and distance measurements.

Fig. 6. Support vector machine multi-class classifier.
5. Experimental results

5.1. Experiments on the home-brewed data set

The home-brewed data set consists of 300 segments of videos covering six kinds of actions. To simplify the experiment, each action segment is a fixed 50-frame-long clip, which is of two-second duration, from the original 25-fps shotted videos. However, the frame-length of action segments is not required to be settled, and it can be selected in the light of the duration of action. The posterior experiments on the public data set exhibit the details.

<table>
<thead>
<tr>
<th>Activity</th>
<th>Amount</th>
<th>Participants</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walk</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>Jogging</td>
<td>53</td>
<td>10</td>
</tr>
<tr>
<td>In-place actions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Standing</td>
<td>18</td>
<td>7</td>
</tr>
<tr>
<td>Sitting</td>
<td>18</td>
<td>7</td>
</tr>
<tr>
<td>Squat</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>Stand-to-sit</td>
<td>51</td>
<td>10</td>
</tr>
<tr>
<td>Stand-to-squat</td>
<td>53</td>
<td>10</td>
</tr>
<tr>
<td>Fall</td>
<td>38</td>
<td>9</td>
</tr>
</tbody>
</table>

The numbers of each activities of our data set are listed in Table 2. Typical images for each kind of activities are shown in Fig. 7. Among the six kinds of activities, the abnormal activity, fall, is paid special attention to, since for those living-alone elders, one great danger is from the fall incident, which is the primary cause of death or other serious diseases. In order to achieve intelligent cares for the elderly, an alarm should be issued in good time when fall happens.

5.1.1. The best feature selection for each SVMs

During the growing of the SVM-BTA, there are several feature combinations on each node under which the best clustering results can be obtained. In the same way, the classification performance of the SVM classifier is influenced by the employed features. And on the other hand, the classification spaces of the K-means clustering and the SVM classifier are different, which possibly makes the best features different. Therefore, the best features of the SVM should be reselected. Besides, considering the computation cost the linear kernel function of the SVM is employed, which has been testified effective by the experimental results.

The process of the features selection for each SVMs is described as follows. First, the total 300 samples are evenly put into two sets, named Set1 and Set2, each of which has 150 samples. Then, two
steps are taken to choose the best features. The first step is to use the samples in Set1 to decide the feature candidates, and the second is to use the samples in Set2 to determine the best features. On each node of the SVM-BTA, half of the samples in Set1 are randomly selected to train the SVM classifier under different features, and the remainders of the Set1 are used to test the classifier, of which the correct recognition rate is considered as the selection criteria of the feature candidates. And furthermore, the best features are determined by the testing results of the samples in Set2. Take SVM1 as an example again to illustrate the process of the features selection. The first step is to decide the feature candidates with the samples of the Set1, and the pending samples on this node consist of 74 samples. The mean error recognition rate of 50 rounds of experiments under the corresponding features is shown in Table 3, which shows that the values of the two kinds of feature \( h \) and \( R \) are too great. Thus, the features CCMEI, the combination of CCMEI and \( h \), and the combination of CCMEI and \( R \) are preselected as the candidates, on which each of the SVM1 classifier is trained. The second step is to select the best features in terms of the recognition performance on the samples of the Set2. Two indicators, the classification accuracy \( A \) and the integrated classification rate, \( F \) defined on the precision \( Pr \) and recall \( Re \) (Yang and Pedersen, 1997), are introduced to measure the performance of each SVM classifiers. Supposing that there are two sorts \( X \) and \( Y \), the numbers of the samples correctly classified of \( X \) and \( Y \) are \( a \) and \( c \), respectively, and the numbers of the samples wrongly classified are \( b \) and \( d \). Then the classification accuracy is defined as:

\[
A = \frac{a + c}{a + b + c + d}
\]  

(13)

In order to obtain an impersonal evaluation of the classification accuracy, 50 rounds of experiments are performed to give a classification accuracy interval in terms of the mean and variance. Table 4 shows the results of each SVMs under the feature candidates. Another indicator is \( F \) defined as:

\[
F = \frac{2 \cdot Pr_x \cdot Re_x}{Pr_x + Re_x}
\]

(14)

where \( Pr_x = \frac{a}{a+b} \) and \( Re_x = \frac{c}{c+d} \). The integrated classification rate curves of 50 rounds of experiments are exhibited in Fig. 8. In the light of the classification accuracy and the integrated classification rate of the SVM1 classifier, the feature combination of CCMEI and \( h \) is considered as the good features for SVM1.

In the same way, take the feature \( h \) for SVM2 and SVM3, the combination of CCMEI and \( R \) for SVM4 and SVM5. Eventually, the SVM-BTA is determined by the certain structure and the right features for each node.

5.1.2. Training and test for SVM-BTA classifier

Randomly select one quarter of the home-brewed data set to train the SVM-BTA classifier, and each SVM is trained separately. Except for the training samples, the remainder samples are subjected to test. Confusion matrix is a good measurement for the overall performance in the multi-class classification problem. The row tab stands for the actual label and the column tab for the recognition label. The values in the diagonal show the recognition accuracy of each class. Table 5 shows the confusion matrix of the SVM-BTA classifier, which indicates that the correct recognition rates for each actions are all more than 90%.

5.2. Experiments on the public database

The database reported in (Schüldt et al., 2004) is a large challenging human action recognition database, which is publicly available. It contains six kinds of actions (walk, jogging, running, boxing, hand waving, and hand clapping) performed by 25 subjects in four different scenarios. Table 6 gives the descriptions of the four scenarios. Some examples of the six types of human action are shown in Fig. 9. The training set used in this paper is the same with Schüldt et al. (2004) and Meng and Pears (2009) which consists of the actions of 8 persons, and the remained action samples of the database constitute the test set.

A recognition system with the SVM-BTA classifier is designed for the six actions of Schüldt’s database to evaluate our system framework further. Since the scene is relatively static, non-param- eter background modelling and updating approach is substituted by simple background estimation method. The background image is estimated as the mean of 3–5 image frames without foreground. The moving object is then obtained as the result of area filtering on the subtraction between the current image and the background image. In the training process, different frame-lengths are chosen for different actions to deal with the various durations. According to the apriori knowledge of the public database, we choose 15 frames for the running action, 30 frames for the jogging and hand clapping actions, 45 frames for the walk, boxing and hand waving actions. However, it should be noted that the frame-length of a testing sample is not appointed but determined by its duration. In real applications, different types of action are progressing in succession and the transitions between them are indetectable. Each type of action can be segmented from long activity sequence by temporal segmentation (Ali and Aggarwal, 2001) firstly, and then the frame-length can be determined by its duration. In real applications, different types of action are progressing in succession and the transitions between them are indetectable. Each type of action can be segmented from long activity sequence by temporal segmentation (Ali and Aggarwal, 2001) firstly, and then the frame-length can be determined by its duration.

After the architecture determination and the best feature selection, an SVM-BTA classifier accompanied by the selected features is obtained as shown in Fig. 10. The confusion matrix of our SVM-BTA classifier is shown in Table 7, which indicates the good recognition performance.

The mean recognition performance is a measurement of the overall classification accuracy. Table 8 lists the mean recognition accuracy of multiple approaches on Schüldt’s database. The table shows that the recognition accuracy of the SVM-BTA in this paper is the best. Among the table, the "SVM + CCMEI in this paper"
indicates a one-against-one SVM multi-classifier with the only feature CCMEI. The recognition accuracy of “SVM + CCMEI” is bigger than that of some other features but smaller than that of SVM-BTA classifier. Thus, the recognition performance of CCMEI is better than some other features and the hierarchical classification thought contributes to improve the recognition performance. In addition, it should be noted that Yeo et al. (2006) attain a relatively high recognition performance but excluding the difficult part (scenarios s2) of the database. The recognition accuracies of the other methods on the whole database are all around 80%, while that of ours SVM-BTA is 88.69%.

Experiments on the homemade data set and the public database both show the efficiency of the prosed recognition system. It should be noted that the actions “jogging” and “running” in the public database are not well discriminated as they are similar actions performed at different speeds.
Table 5
Confusion matrix of the SVM-BTA classifier.

<table>
<thead>
<tr>
<th>Actions</th>
<th>In-place actions</th>
<th>Walk</th>
<th>Jogging</th>
<th>Stand-to-sit</th>
<th>Stand-to-squat</th>
<th>Fall</th>
</tr>
</thead>
<tbody>
<tr>
<td>In-place</td>
<td>100</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Walk</td>
<td>0</td>
<td>95</td>
<td>5</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jogging</td>
<td>0</td>
<td>7</td>
<td>93</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Stand-to-sit</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>94.6</td>
<td>5.4</td>
<td>0</td>
</tr>
<tr>
<td>Stand-to-squat</td>
<td>0</td>
<td>0</td>
<td>2.4</td>
<td>97.6</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Fall</td>
<td>0</td>
<td>7.8</td>
<td>0</td>
<td>0</td>
<td>92.2</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 6
Descriptions of the four types of scenarios in the database (Schüldt et al., 2004).

<table>
<thead>
<tr>
<th>Scenarios mark</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>s1</td>
<td>Outdoors, camera is parallel to the object moving trajectories</td>
</tr>
<tr>
<td>s2</td>
<td>Outdoors, some angle between the camera and the object moving trajectories, or scale changes</td>
</tr>
<tr>
<td>s3</td>
<td>Outdoors, different clothes or pack on the back</td>
</tr>
<tr>
<td>s4</td>
<td>Indoors, some or even heavy shadows</td>
</tr>
</tbody>
</table>

6. Conclusion

The recognition of human activities has gained increasing interest in computer vision, while it remains challenging owing to the complexity and variety of human actions. In this paper, a new system framework for recognizing multiple human activities from video sequences by SVM-BTA is proposed. The framework is composed of three cascaded modules including people detection and location, feature vector extraction from human blobs, and activity recognition. A new type of contour feature named CCMEI is firstly presented. Moreover, the SVM multi-class classifier with binary tree architecture, whose structure is determined by clustering, is firstly employed to recognize human activities.

A recognition system is constructed for the homemade data set, which includes walk, jogging, in-place actions, stand-to-sit, stand-to-squat, and fall, to implement intelligent home-care. And another system is constructed for Schüldt’s public data set, which includes walk, jogging, running, boxing, hand clapping and hand waving, to evaluate the performance of the proposed feature CCMEI and the classifier SVM-BTA. Experimental results of the two systems both show the efficiency and effectiveness of the system framework in this paper.

In the proposed system framework the method of object detection can be chosen in terms of the scene. The non-parameter background subtraction approach can be substituted by other simple person detection algorithm if the scene is relatively static. Thus, the computation cost of our system is divided into two parts.

One is the cost of object detection process, and the other is the cost addition of feature extraction and activity classification, which is referred to as the classification cost. The object detection by non-parameter background subtraction plus postprocessing used in this paper can achieve about 15 frames per second on a 2.8 GHz Pentium processor for 320 x 240 RGB image. The classification costs of the SVM-BTA classifiers of the homemade data set and Schüldt’s database are counted separately. The mean classification cost of each action in our data set is about 0.8872 s while that in Schüldt’s database is about 0.2475 s. It should be noted that the image size in

Table 7
Confusion matrix of the SVM-BTA classifier on Schüldt’s database.

<table>
<thead>
<tr>
<th>Actions</th>
<th>Walk</th>
<th>Jog</th>
<th>Run</th>
<th>Box</th>
<th>Clap</th>
<th>Wave</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walk</td>
<td>96.43</td>
<td>1.19</td>
<td>0</td>
<td>2.38</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Jog</td>
<td>17.86</td>
<td>82.14</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Run</td>
<td>1.19</td>
<td>26.19</td>
<td>72.62</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Box</td>
<td>0</td>
<td>0</td>
<td>95.24</td>
<td>3.57</td>
<td>1.19</td>
<td>0</td>
</tr>
<tr>
<td>Clap</td>
<td>0</td>
<td>0</td>
<td>1.19</td>
<td>98.81</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Wave</td>
<td>0</td>
<td>0</td>
<td>2.38</td>
<td>10.71</td>
<td>86.91</td>
<td>0</td>
</tr>
</tbody>
</table>

Table 8
Comparison of different methods on Schüldt’s database.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Recognition accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SVM-BTA in this paper</td>
<td>88.69</td>
</tr>
<tr>
<td>SVM + CCMEI in this paper</td>
<td>79.57</td>
</tr>
<tr>
<td>SVM + LF (Schüldt et al., 2004)</td>
<td>71.72</td>
</tr>
<tr>
<td>Cascade of filters + Volumetric features (Ke et al., 2005)</td>
<td>62.97</td>
</tr>
<tr>
<td>SVM + MGD and Hist. of MHI (Meng and Pears, 2009)</td>
<td>80.3</td>
</tr>
<tr>
<td>SVM + spatiotemporal features (Dollár et al., 2005)</td>
<td>81.17</td>
</tr>
<tr>
<td>pLSA + spatial-temporal words (Niebles et al., 2008)</td>
<td>83.33</td>
</tr>
<tr>
<td>KNN + NZMS (Yeo et al., 2006)</td>
<td>86</td>
</tr>
</tbody>
</table>

Fig. 9. Examples of the six types of human action in Schüldt’s database.

Fig. 10. Support vector machine multi-class classifier for the actions in Schüldt’s database.
our data set is $320 \times 240$ while that in Schüldt’s database is $120 \times 160$.

Further research on this work is to construct a multi-camera surveillance system on which long-term video surveillance would be performed and the proposed activity recognition algorithm could be further tested.

Acknowledgment

The research is partially supported by the State Nature Science Foundation under Grant 60974129, 70931002, and the open-end fund of Jiangsu Provincial Key Laboratory under Grant KJS08020.

References


