International Journal of Control and Automation
Vol.9, No.3 (2016), pp.245-260
http://dx.doi.org/10.14257/ijca.2016.9.3.23

Action Recognition Using Hierarchical STIP Saliency and Mixed Neighborhood Features

Jiangfeng Yang and Zheng Ma

School of Communication and Information Engineering
University of Electronic Science and Technology of China, Xiyuan Ave,
No.2006, West Hi-Tech Zone, 61173
wallsonyang@163.com, 369322023@qq.com

Abstract

In video action recognition, the Dollar detector has been widely used to extract Spatio-Temporal Interest Points (STIPs) from action video sequence. It generates two kinds of information: STIP position and the respond value. However, in many cases, the detector respond, which measures the strength of local motion changes, is ignored. By utilizing such information, we propose to build a Hierarchical STIP Saliency (HSS) framework to provide different types of motion information. A novel local feature named Mixed Neighborhood Feature (MNF), which integrates the similarity and position relationship between local features, is put forward, and encoded by locality-constrained linear coding. Then, by partitioning video sequence along temporal direction, a group of sub-STVs are produced, and their corresponding descriptors are obtained with a max-pooling-on-absolute-value technique. In classification stage, Locality-constrained Group Sparse Representation (LGSR) is adopted as classifier to utilize the intrinsic group information of these sub-STV features. The experiments on the KTH and UCF Sports datasets show that in contrast to the classical recognition systems published recently, our recognition system based on the HSS and MNF achieves good performance.

Keywords: *Action recognition, hierarchical STIP saliency, mixed neighborhood features, action representation*

1. Introduction

Automatic image categorization has drawn increasing attention of the researchers around the world due to its widespread prospects in various applications (e.g., video surveillance, image and video retrieval, web content analysis, and biometrics). In recent works dealing with the image categorization tasks, the bag-of-features (BoF) based model, developed from the bag-of-words model in document analysis, has been proved to be an efficient models in addressing this problem, where the local features are quantized to form a visual vocabulary, and a video sequence is summarized by the histogram of its feature occurrences. The representation has a number of advantages: being local, the features have robustness to viewpoint changes and partially occlusions; being relatively sparse, they can be stored and manipulated efficiently.

However, a key limitation of spatio-temporal interest point (STIP) representations is that they can be too local, fail to capture adequate spatial or temporal relationships. In the extreme, the orderless BoF lacks cues about motion trajectories, before-after relationships, or the relative layout of objects and actions. In an attempt to overcome this problem, several alternatives [1-7] have been proposed to capture the spatio-temporal (ST) relationship between local features. Among these approaches, $\varepsilon-$ neighborhood and KNN-neighborhood

ISSN: 2005-4297 IJCA

(NN, nearest neighbor) are the most popular ones. In ε–neighborhood method, if the distance between two features is less than a threshold ε, where ε denotes the radius of the local neighborhood, the two features are defined as "close". However, fixed radius ε leads this method to fail to adapt to the scale changes. In KNN-based neighborhood method, the neighborhood consists of its nearest neighbors of a local feature; and the major limitation of this method lies in that the relative position information between neighbors (e.g., before/after, above/below, left/right) is not taken into account.

To handle the two limitations, we propose to construct a mixed neighborhood feature (MNF) (see Figure 2), which not only adapts to the scale change, but also combines the relative position and similarity information between the neighboring features.

In addition, to extract different types of motion information for action representation, a multi-level STIP framework named Hierarchical STIP Saliency (HSS) is built. As we know, the saliency of a spatial-temporal position can be measured by its respond value of STIP detector. In building HSS, by setting a STIP-number threshold *N* to the STIP detector, the STIPs corresponding to the first *N* greatest responds are selected to form one level in HSS.

Undergoing the above manners, an action video is represented as a collection of *N* MNFs. To reduce quantization error in the stage of encoding MNFs, Locality-constrained Linear Coding (LLC) [8] algorithm is employed, and the related reconstruction coefficients are obtained. Next, since a video sequence can be regarded as a ST volume (STV), in order to capture the ST distribution of MNFs within a STV, the STV is partitioned into several sub-STVs along temporal axis. Finally, their sub-STV descriptors are computed by max-pooling-on-absolute-value [9] method upon the reconstruction coefficient vectors.

In classification, Locality-constrained Group Sparse Representation (LGSR) [10] is used as action classifier. The experiments on the KTH and UCF Sports datasets show that our method achieves better performance than the classical methods published recently [11-16].

In the paper, two contributions are made as follows:

- To extract more helpful motion information and make use of the respond value of STIP, a novel multi-layer framework based on STIP saliency is constructed for improving the performance of action recognition system
- To solve the limitations of traditional neighborhood feature, a new neighborhood description method named MNF is proposed. It combines the similarity and position relationship between local features, and handles the problem of scale changes.

The rest of this paper is organized as follows: Constructing HSS with the extracted STIPs is proposed in Section 2. Section 3 provides the formation of MNF. Then, encoding the obtained MNFs by LLC algorithm is shown in Section 4. And Section 5 presents that generating multi-temporal-scale sub-STVs, computing their corresponding descriptors, and classifying actions with LGSR method. Experimental results and analysis are shown in Section 6. Finally, conclusions are drawn in Section 7.

2. Constructing Hierarchical STIP Saliency (HSS)

Above all, the inputs to our recognition system are the STIP positions and their associated local descriptors. We utilize Dollar detector [17] to extract STIPs form video sequences. The detector generally produces a great number of STIPs that is important for constructing HSS. The response function of Dollar detector has the form

$$R(x,y,t) = (V(x,y,t) * h_{ev})^2 + (V(x,y,t) * h_{od})^2,$$
$$(x = 1,\cdots,V_{length};\ y = 1,\cdots,V_{width};\ t = 1,\cdots,V_{frame}), \quad (1)$$
$$h_{ev}(t;\tau,\omega) = -\cos(2\pi\omega t)\exp(-t^2/\tau^2); \qquad h_{od}(t;\tau,\omega) = -\sin(2\pi\omega t)\exp(-t^2/\tau^2)$$

where $g(x,y;\sigma)$ is the 2D Gaussian smoothing kernel, applied only along the spatial dimension, and h_{ev} and h_{od} are a quadrature pair of 1D Gabor filters applied temporally, two parameters σ and τ corresponds to spatial and temporal scale of the detector, respectively. $V_{length}, V_{width}, V_{frame}$ denote the length, width, and frame number of action video V. Unlike authors in [17] used single-scale STIP detector that is not robust to environment noise, we use multi-scale Dollar detector to extract reliable STIPs from action video. More detailed, the final respond value at each position equals to the sum of several single-scale responds at this position.

Given an action video V, by setting a STIP-number threshold $N_l (l = 1,\cdots,L)$ to the Dollar detector, the l-th level of HSS is built. The constructed HSS is expressed as follows:

$$HSS(V) = \{SL_l : l = 1,\cdots,L\},$$
$$SL_l = select(\{R(x,y,t)\}, N_l, V) \quad (2)$$
$$= \{(x_i, y_i, t_i)^l : i = 1,\cdots,N_l\}$$

where $(x_i, y_i, t_i)^l, (x = 1,\cdots,V_{length}; y = 1,\cdots,V_{width}; t = 1,\cdots,V_{frame})$ denotes the coordinate of the selected i-th STIP in level-l; $R(x,y,t)$ is the respond value at position (x,y,t); the function $select(.,N_l,V)$ is to select the N_l STIPs corresponding to the largest respond values to build level-l of HSS; $N_a \le N_b$, if $a \le b$ (see Figure 1).

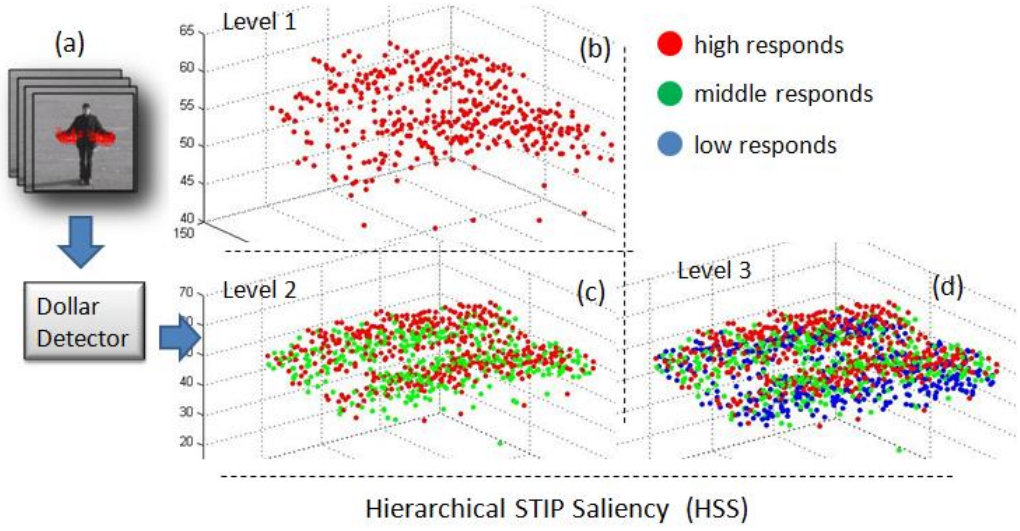


Figure 1. A Constructed Three-Level HSS for Action "Hand-Clapping" in the KTH Dataset. In HSS, from the Top Level to The Bottom Level, their STIP Numbers Increase. (A) Input Video Sequence. (B) The Level-1 (Top Level) Stips have High Responds (Red Circle) of Dollar Detector. (C) The Level-2 Stips have Middle Responds (Green Circle). (D) The Level-3 (Bottom Circles) Stips have Low Responds (Blue Circles)

By setting a small threshold N_a and a large threshold N_b to the Dollar detector ($N_a \le N_b$, if $a \le b$), a high level and a low level in HSS are created, respectively. The STIPs in the high level have strong responds of the Dollar detector, which implies that the motion information at theses location changes dramatically along temporal axis; on the other hand, the responds of STIPs in the low level change greatly, which means that some with low responds could be noisy data.

As a result, the different levels in HSS provide different types of information for action representation: the high levels present the fundamental, reliable motion information about human action; the low levels offer a great amount of detailed, less reliable motion information. It could be concluded that the motion information from the high/low levels in HSS is supplementary to each other.

3. Forming Mixed Neighborhood Features (MNF)

Once obtaining HSS, local features (e.g., HOG, histogram of oriented gradient, and HOF, histogram of optical flow [2]) at STIPs are computed, and HSS can be rewritten as follows

$$\begin{aligned} &HSS(V) = \{SL_l : l = 1, \cdots, L\}, \\ &SL_l = \{(x_i, y_i, t_i)^l, \mathbf{d}_i^l : i = 1, \cdots, N_l\}, \end{aligned} \tag{3}$$

where $\mathbf{d}_i^l$ denotes local feature at location $(x_i, y_i, t_i)^l$.

To learn the MNF of STIP p, we firstly build its KNN-based neighborhood, which consists of several nearest neighbors around p, where nearness is measured by a Euclidean distance on its 3d position coordinates. Let $N(p) = \{p, q_1, \cdots, q_{nb}, \cdots, q_{NB-1}\}$ denote the NB nearest neighboring STIPs for p, where $q_{nb} = (x_{nb}, y_{nb}, t_{nb}, \mathbf{d}_{nb}), p = (x, y, t, \mathbf{d})$, and $\mathbf{d}_{nb}, \mathbf{d}$ denote their local feature descriptors. The angle $\theta_{nb}, (0 \le \theta_{nb} \le 180)$ between $\mathbf{d}_{nb}$ and $\mathbf{d}$ is computed as their similarity, and quantized into K levels by $\lfloor \theta_{nb} / (180/(K-1)) \rfloor$. Additionally, each neighbor can be placed in one of 2×2×2=8 direction bins, depending on its location in time and space with respect to the central STIP — to the left or right, below or above, before or after (see Figure 2).

To describe the information about neighborhood $N(p)$, we form a matrix $\mathbf{M}$ with size 8-by-K (8 directions, K angle-levels). Specifically, $\mathbf{M}$ is built based on the NB feature descriptors $\{\mathbf{d}, \mathbf{d}_1, \cdots, \mathbf{d}_{NB-1}\}$. The entries in $\mathbf{M}$ record how many of the neighbors fall into each of the direction-angle bins (see Figure 3). In other words, for each neighboring STIP q_{nb}, we increment bins according to both of its position relative to the central STIP p and the angle θ_{nb}.

To extract more distribution information about the neighborhood $N(p)$, R matrixes $\{\mathbf{M}_r\}_{r=1}^R$ are created by setting the different neighbor number $\{NB_r\}_{r=1}^R$ and repeating the above procedure. Each matrix is reshaped to a $8K$-dimensional vector, and ℓ_1 normalized. Next, all the R normalized vectors are concatenated on top of each other to form a single neighborhood descriptor $\mathbf{f} \in \Re^{8KR}$, which is the proposed MNF of p. Finally, the HSS of a video sequence V can be represented as follows:

$$\begin{aligned} &HSS(V) = \{SL_l : l = 1, \cdots, L\}, \\ &SL_l = \{(x_i, y_i, t_i)^l, \mathbf{d}_i^l, \mathbf{f}_i^l : i = 1, \cdots, N_l\}, \end{aligned} \tag{4}$$

where $\mathbf{f}_i^l$ denotes the MNF feature corresponding to the i-th STIP at level-l of HSS.

Note that since the local features are selected into the neighborhood according to their distance from the central STIP. Hence, the neighborhood descriptor MNFs can keep the property of scale-invariant from scale changes.

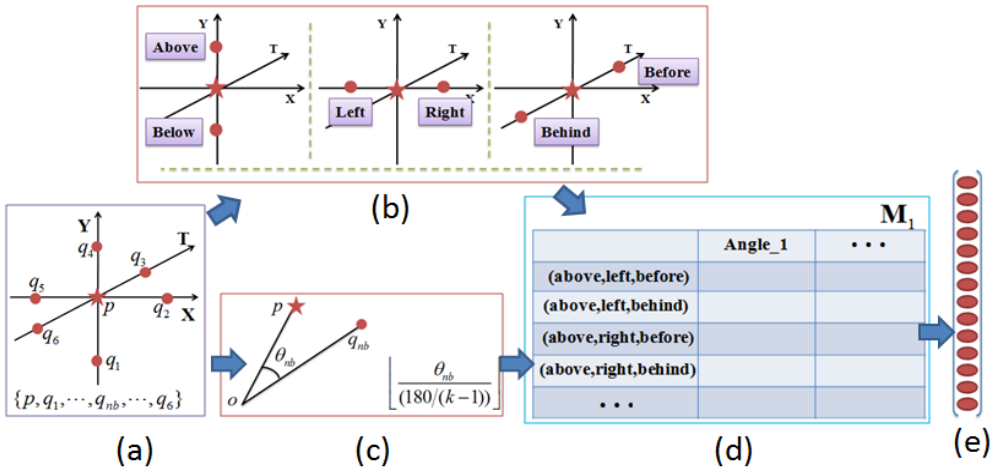


Figure 2. The Construction of MNF. (A) A Neighborhood Consists of 7 Nearest Neighbors. (B) The Relative Position Relationship between Neighbors and Central STIP . (C) The Cosine Angle between and Quantized. (D) A Matrix is Generated to Store the Neighborhood Information. (E) A Feature Vector is Created by Reshaping the Matrix

4. Encoding Mnfs By Locality-Constrained Linear Coding (LLC)

Let $\{\mathbf{f}_i^l : i = 1, \cdots, N_l\}$ be N_l MNF features at level-l of HSS. A codebook with M bases $\mathbf{B}^l = [\mathbf{b}_1^l, \mathbf{b}_2^l, \cdots, \mathbf{b}_M^l] \in R^{8KR \times M}$, for simplicity, is generated by k-means clustering over training samples with Euclidean distance as metric. Feature $\mathbf{f}_i^l$ is converted into an M-dimensional code $\mathbf{c}_i^l \in R^M$ by feature coding schemes, such as vector quantization (VQ), soft vector quantization (SVQ), localized soft assignment (LSVQ), sparse coding (SC), and LLC. We briefly review these coding methods in the next section.

4.1. VQ, SVQ and SC

In VQ, its coding strategy assigns just a single base to a local feature; each local feature is assigned to the nearest visual codeword:

$$c_{i,j}^l = \begin{cases} 1, & \text{if } j = \arg\min_j \left\| \mathbf{f}_i^l - \mathbf{b}_j^l \right\|_2^2, \\ 0, & \text{otherwise,} \end{cases} \tag{5}$$

where the resulting code $\mathbf{c}_i^l = [c_{i,1}^l, c_{i,2}^l, \cdots, c_{i,M}^l]^T$. This coding is simple but, as reported in [8], suffers from the reconstruction error due to the reason that it only assigns a single code word to the descriptor.

To alleviate the quantization error of VQ, Gemert *et al.* [18] proposed SVQ on which a feature is encoded across several codewords instead of using one:

$$c_{i,j}^{l} = \frac{\exp(-\beta \left\| \mathbf{f}_i^l - \mathbf{b}_j^l \right\|_2)}{\sum_{m=1}^{M} \exp(-\beta \left\| \mathbf{f}_i^l - \mathbf{b}_m^l \right\|_2)}, \quad (6)$$

where β is a parameter controlling how widely the assignment distributes the weight across all the code words. A small β gives a broad distribution, while a large β gives a peaked distribution, more closely approximating hard assignment.

SVQ is further improved by Liu *et al.* [19], who used localized soft assignment (LSVQ). Instead of distributing the weights across all codebook elements, they confine the soft assignment to a local neighborhood around the descriptor being coded. Let $NN_{(S)}(\mathbf{f}_i^l)$ be the set of S nearest neighbors to $\mathbf{f}_i^l$ in $\mathbf{B}^l$. Then, the localized soft assignment coding is:

$$c_{i,j}^{l} = \frac{\exp(-\beta d(\mathbf{f}_i^l, \mathbf{b}_j^l))}{\sum_{m=1}^{M} \exp(-\beta d(\mathbf{f}_i^l, \mathbf{b}_m^l))}, \quad (7)$$

$$d(\mathbf{f}_i^l, \mathbf{b}_j^l) = \begin{cases} \left\| \mathbf{f}_i^l - \mathbf{b}_j^l \right\|_2^2, & \text{if } \mathbf{b}_j^l \in NN_{(s)}(\mathbf{f}_i^l) \\ \infty, & \text{otherwise} \end{cases}, \quad (8)$$

Another way to reduce the quantization loss of VQ is SC [8] that encodes a local feature by using the coefficients of a linear combination of the codewords in $\mathbf{B}^l$, with a sparsity regularity term ℓ_1-norm:

$$\mathbf{c}_i^l = \arg\min_{\mathbf{c}} (\left\| \mathbf{f}_i^l - \mathbf{B}^l \mathbf{c} \right\|_2^2 + \lambda \left\| \mathbf{c} \right\|_1), \quad \lambda \in R, \quad (9)$$

where the first term represents the reconstruction error of $\mathbf{f}_i^l$ with respect to codebook $\mathbf{B}^l$. The second term denotes a sparse constraint regularization on code $\mathbf{c}$, and λ is a regularization factor to balance these terms.

Although compared to VQ, SC significantly reduces the quantization loss, its computation complex is high, and not guarantee that same input features produce same encoding result.

4.2. Locality-Constrained Linear Coding (LLC)

In contrast to the previous coding schemes, LLC coding algorithm has attracted much attention due to its impressive properties:

- Better reconstruction. In VQ (Figure 3), each descriptor is represented by a single basis in the codebook. Due to the large quantization errors the VQ code for similar descriptors might be very different. Besides, the VQ process ignores the relationships between different bases. Hence non-linear kernel projection is required to make up such information loss. On the other side, as shown in (Figure 3) in LLC, each descriptor is more accurately represented by multiple bases, and LLC code captures the correlations between similar descriptors by sharing bases.
- Local smooth sparsity. Similar to LLC, SC also achieves less reconstruction error by using multiple bases. Nevertheless, the regularization term of norm in SC is not smooth. As (shown in Figure 3), due to the over-completeness of the codebook, the SC process might select quite different bases for similar patches to favor sparsity, thus losing correlations between codes. On the other side, the explicit locality adaptor in LLC ensures that similar patches will have similar

codes.

- Analytical solution. Solving SC usually requires computationally demanding optimization procedures. Unlike SC, the solution of LLC can be derived analytically such that LLC can be performed very fast in practice.

LLC coding scheme bases on the hypothesis that descriptors approximately reside on a lower dimensional manifold in an ambient descriptor space; thus, it reduces the quantization error while preserving the consistent encoding ability.

Unlike the sparse coding, LLC enforces locality instead of sparsity and this leads to smaller coefficient for the basis vectors far away from the local feature $\mathbf{f}_i^l$. The coding coefficients are obtained by solving the following optimization:

$$\mathbf{c}_i^l = \arg\min_{\mathbf{c}}(\left\|\mathbf{f}_i^l - \mathbf{B}^l\mathbf{c}\right\|_2^2 + \lambda\left\|\mathbf{d}^l \odot \mathbf{c}\right\|_2^2), \quad \text{s.t.} \quad \mathbf{1}^T\mathbf{c} = 1, \tag{10}$$

$$\mathbf{d}^l = \exp(\frac{\text{dist}(\mathbf{f}_i^l, \mathbf{B}^l)}{\sigma}), \quad \text{dist}(\mathbf{f}_i^l, \mathbf{B}^l) = [\text{dist}(\mathbf{f}_i^l, \mathbf{b}_1^l), \cdots, \text{dist}(\mathbf{f}_i^l, \mathbf{b}_M^l)]^T, \tag{11}$$

where the first term is reconstruction error; the second term is the locality constraint regularization on code $\mathbf{c}$, and λ is a regularization factor; in the second term, $\odot$ denotes the element-wise multiplication, and $\mathbf{d}^l \in \Re^M$ is the locality adaptor that gives different weight for each base vector proportional to its similarity to the input feature $\mathbf{f}_i^l$; and $\text{dist}(\mathbf{f}_i^l, \mathbf{b}_j^l)$ is the Euclidean distance between $\mathbf{f}_i^l$ and the j-th base $\mathbf{b}_j^l$. Parameter σ is used for adjusting the weight decay speed for the locality adaptor. $\mathbf{1}^T\mathbf{c} = 1$ denotes the shift invariant constraint according to [8].

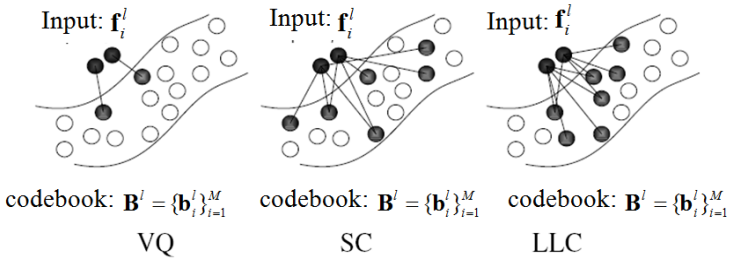


Figure 3. Comparison among VQ, SC and LLC. The Selected Bases for Representation are highlighted in Black

In the paper, we first build a codebook $\mathbf{B}^l$ with M_l bases $\mathbf{B}^l = [\mathbf{b}_1^l, \mathbf{b}_2^l, \cdots, \mathbf{b}_{M_l}^l]$. Then, LLC is employed to encode $\{\mathbf{f}_i^l : i = 1, \cdots, N_l\}$ using $\mathbf{B}^l$, and obtain the corresponding reconstruction coefficient $\{\mathbf{c}_i^l \in \Re^{M_l} : i = 1, \cdots, N_l\}$

5. Classifying Action Videos with LGSR

After converting MNFs $\{\mathbf{f}_i^l : i = 1, \cdots, N_l\}$ into their corresponding coefficients $\{\mathbf{c}_i^l \in \Re^{M_l} : i = 1, \cdots, N_l\}$, action video sequence V is represented as a set of reconstruction coefficient vectors $\{\{\mathbf{c}_i^l \in \Re^{M_l} : i = 1, \cdots, N_l\} : l = 1, \cdots, L\}$ and equation (4) is rewritten as

$$\begin{aligned} &HSS(V) = \{SL_l : l = 1, \cdots, L\}, \\ &SL_l = \{(x_i, y_i, t_i)^l, \mathbf{c}_i^l : i = 1, \cdots, N_l\}, \end{aligned} \tag{12}$$

where $\mathbf{c}_i^l$ denotes the reconstruction coefficient of MNF feature $\mathbf{f}_i^l$; N_l denotes the MNF number in level-*l* of HSS.

5.1. Extracting Sub-STVS with Multi-Temporal-Scale Sampling (MTS)

Due to the different styles of human action, it is difficult to model the ST relationship of local features in a single space-time scale. The actions with different styles appear in different motion range (different spatial scale) and speed (different temporal scale). Fortunately, the ST relationships between local features can be locally modeled by several sub-STVs, which are obtained by dividing a STV along temporal axis. In the paper, to enhance the robustness of action representation to the various action speeds, MTS method is used in extracting sub-STVs.

5.2. Constructing Sub-STV Descriptors

For the *i*-th STIP in level-*l* of HSS, its sub-STV with temporal-scales $\alpha(r),(r \in \{1,\cdots,R\})$ that contains *n* MNF features is produced

$$\mathbf{F}_i^{l,\alpha(r)} = \{\mathbf{f}_1, \mathbf{f}_2, \cdots, \mathbf{f}_n\}, \tag{13}$$

Where *R* is the tempo-scale number in MTS method.

Then, LLC is used to encode each feature and obtain their codes:

$$\mathbf{C}_i^{l,\alpha(r)} = \{\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_n\}, \tag{14}$$

After encoding each feature, pooling techniques are used to the sub-STV descriptor. It is noted that the elements in code $\mathbf{c}_i,(i=1,\cdots,n)$ could be positive or negative value, so that we choose the max-pooling function on the absolute codes [9] as our pooling method, instead of the popular max-pooling [8] and average-pooling [8]. The max-pooling on the absolute codes $|\mathbf{c}_i|,(i=1,\cdots,n)$ and its ℓ_1 normalization are defined as follows:

$$\begin{aligned} \mathbf{s}_i^{l,\alpha(r)}(j) &= \max\{|\mathbf{c}_1(j)|, |\mathbf{c}_2(j)|, \cdots, |\mathbf{c}_n(j)|\}, \\ \mathbf{s}_i^{l,\alpha(r)} &= \mathbf{s}_i^{l,\alpha(r)} \Big/ \sum\nolimits_j \mathbf{s}_i^{l,\alpha(r)}(j) \end{aligned} \tag{15}$$

Where $\mathbf{s}_i^{l,\alpha(r)}(j)$ the j-th element of is $\mathbf{s}_i^{l,\alpha(r)}$, $\mathbf{c}_i(j)$ is the *j*-th element of code $\mathbf{c}_i$. This max pooling procedure is well established by biophysical evidence in visual cortex (V1) [20] and is empirically justified by many algorithms applied to image categories.

Using MTS method for $\mathbf{f}_i^l$, *R* sub-STVs with temporal-scales $\alpha(r),(r=1,\cdots,R)$ are generated. Then, repeating the above manner upon each sub-STV, and obtain pooled and normalized codes $\{\mathbf{s}_i^{l,\alpha(r)}(j)\}_{r=1}^R$. Finally, the sub-STV descriptor $\mathbf{s}_i^l$ is obtained by concatenating all codes $\{\mathbf{s}_i^{l,\alpha(r)}(j)\}_{r=1}^R$:

$$\mathbf{s}_i^l = [\mathbf{s}_i^{l,\alpha(1)}, \mathbf{s}_i^{l,\alpha(2)}, \cdots, \mathbf{s}_i^{l,\alpha(R)}], \tag{16}$$

Where $\mathbf{s}_i^l$ is the sub-STV descriptor of the *i*-th STIP in the *l*-th level.

5.3. LGSR

To utilize the intrinsic group information from these sub-STV descriptors within one video for action classification, we adopt the Locality-constrained

Group Sparse Representation (LGSR) for action classification task. LGSR was proposed in [10] for human gait recognition. It is an extended Sparse Representation-based Classifier (SRC). The pioneering work of SRC was proposed in [21] and used to classify face images by minimizing the norm-regularized reconstruction error (RE). Compared with SRC, LGSR has three advantages:

- SRC is designed for single image classification and cannot directly classify a group of samples, while LGSR is designed for sample group classification.
- The locality constraint in LGSR is more reasonable than sparsity constraint in SRC, especially for representing manifold data [22, 23].
- LGSR is a block sparse constraint classifier. It is better than SRC in classification task when the used features are discriminative.

The object function of original LGSR is defined as:

$$\mathbf{C}^* = \arg\min_{\mathbf{C}}\left(\frac{1}{2}\left\|\mathbf{Y}-\mathbf{B}\mathbf{C}\right\|_F^2 + \lambda\sum_{k=1}^{K}\left\|\mathbf{D}^k \square\ \mathbf{C}^k\right\|_F\right), \tag{17}$$

Where the first term represents RE of the test action $\mathbf{Y}$ with respect to the training dictionary $\mathbf{B}$ built upon sub-STV descriptors. The second term is the weighted $\ell_{1,2}$ mixed-norm-based regularization on the reconstruction coefficient $\mathbf{C}^k$. λ is the regularization parameter to balance these terms.

In our system, the RE is produced by each level of HSS, and should be taken into account. So, the LGSR is revised as follows:

$$\begin{aligned}\{(\mathbf{C}^l)^* : l=1,\cdots,L\} &= \arg\min_{\{\mathbf{C}^l : l=1,\cdots,L\}}\sum_{l=1}^{L}\left(\frac{1}{2}\left\|\mathbf{Y}^l-\mathbf{B}^l\mathbf{C}^l\right\|_F^2 + \lambda_l\sum_{k=1}^{K}\left\|\mathbf{D}_k^l \square\ \mathbf{C}_k^l\right\|_F\right),\\ &= \sum_{l=1}^{L}\left(\arg\min_{\mathbf{C}^l}\left(\frac{1}{2}\left\|\mathbf{Y}^l-\mathbf{B}^l\mathbf{C}^l\right\|_F^2 + \lambda_l\sum_{k=1}^{K}\left\|\mathbf{D}_k^l \square\ \mathbf{C}_k^l\right\|_F\right)\right)\end{aligned} \tag{18}$$

Where the first and second terms are RE and regularization constraint with respect to each level of HSS, respectively; $\mathbf{Y}^l$ denotes the sub-STV descriptors at the level-l for one test action; $\mathbf{B}^l$ is classification dictionary constructed by connecting K class-specific dictionaries $[\mathbf{B}_1^l,\cdots,\mathbf{B}_K^l]$. Each class-specific dictionary $\mathbf{B}_k^l$ is learnt with K-means algorithm over the sub-STV descriptors, which belong to the level-l HSS and correspond to the k-th action class; $\mathbf{C}^l$ is reconstruction coefficient and corresponds to $\mathbf{B}^l$, and $\mathbf{C}^l=[\mathbf{C}_1^l,\cdots,\mathbf{C}_K^l]$; $\mathbf{D}_k^l$ is the distance matrix between $\mathbf{Y}_k^l$ and $\mathbf{B}_k^l$, and the entry $\mathbf{D}_k^l(i,j)=\left\|\mathbf{Y}_k^l(i)-\mathbf{B}_k^l(j)\right\|_2$; λ_l is the regularization parameter.

It can be found in (18) that when calculating the level-l RE, $\mathbf{C}_k^l$ values are independent to each other, we can separately update each $\mathbf{C}_k^l$ using its subgradient [23]. To solve (18), the active set-based subgradient descent algorithm in [10, 24] was employed.

5.4. Classification Methods

Once we obtain the optimal reconstruction coefficients $\{(\mathbf{C}^l)^* : l=1,\cdots,L\}$, we can use two classification methods [10] based on different criteria to classify the test video.

1) Minimum Reconstruction Error (minRE) criterion: We compute the reconstruction error for each class as follows:

$$R_k(\{(\mathbf{C}^l)^*\}_{l=1}^L) = \frac{1}{2}\sum_{l=1}^{L}\beta_l \left\| \mathbf{Y}^l - \mathbf{B}_k^l (\mathbf{C}_k^l)^* \right\|_F, \quad (19)$$

where the reconstruction coefficient $(\mathbf{C}_k^l)^*$ is related to the level-*l* in HSS of the *k*-th training video; β_l is the weight for the level-*l* RE. Then, we classify the test video to $k^* = \arg\min_k R_k(\{(\mathbf{C}^l)^*\}_{l=1}^L)$, as in [10].

2) Maximum Weighted Inverse Reconstruction Error (maxWIRE) criterion: In the above criterion, the reconstruction coefficient is not used directly for classification. Intuitively, if the reconstruction errors of the test video with respect to two training videos are the same, we should choose the class label of the training video that is associated with the larger Frobenius norm of the reconstruction coefficient. Specifically, we define the following weighted inverse reconstruction error

$$Q_k(\{(\mathbf{C}^l)^*\}_{l=1}^L) = \frac{\sum_{l=1}^{L}\beta_l \left\| (\mathbf{C}_k^l)^* \right\|_F}{\sum_{l=1}^{L}\beta_l \left\| \mathbf{Y}^l - \mathbf{B}_k^l (\mathbf{C}_k^l)^* \right\|_F}, \quad (20)$$

where β_l is the weight of the level-*l* RE. Then, we classify the test video to $k^* = \arg\max_k Q_k(\{(\mathbf{C}^l)^*\}_{l=1}^L)$,

In the paper, we use maxWIRE criterion as human video action classifier.

6. Experiments

In this section, two public video datasets, the KTH and UCF sports datasets, are used to evaluate the performance of our recognition system based on HSS and MNF.

6.1. Experimental Configuration

The experimental configuration on the two datasets is as follows:

In all experiments, Dollar detector based on multiple ST scales is used to extract STIPs from action videos, and its spatial scale $\tau = [1.2, 1.3, 1.4, 1.5]$, temporal scale $\omega = [0.4, 0.45, 0.5, 0.55]$, and HOG/HOF [2] is adopted to describe these STIPs; the level number of HSS is set to 3 $(L = 3)$, and the STIP number threshold N_l is related to the number of frames that contain human motion: for the KTH, $N_1 = (MF \times 1.0), N_2 = (MF \times 2.0)$ and $N_3 = (MF \times 2.5)$; for the UCF Sports, $N_1 = (MF \times 1.0), N_2 = (MF \times 1.5)$, and $N_3 = (MF \times 2.0)$, where *MF* denotes there exists *MF* frames containing human action.

In building MNF features, two-level neighborhood $(R = 2)$ is involved. For the KTH, its level-1, 2 neighborhood consists of 5, 7 neighbors $(NB_1 = 5, NB_2 = 7)$, respectively. For the UCF Sports, $NB_1 = 3, NB_2 = 5$.

To obtain the dictionaries for LLC coding in training stage, features from 24 videos (4 videos per action, 6 actions) of one subject are clustered by k-means for the KTH; and features from 20 videos (2 videos selected from each action, 10 actions) are clustered by k-means for the UCF Sports. The resulting dictionary size is set to 400 for KTH, and 500 for UCF Sports. The number of selected bases is set to 5 in encoding MNF features.

To capture multi-scale spatial relationship of local features, the lengths of sub-STV are set to 5, 10, 15, and 20 frames. Since there are 4 spatial scales, the dimension of a sub-STV descriptor, which is concatenated by 4 pooled codes, is $400 \times 4 = 1600$ for the KTH, and $500 \times 4 = 2000$ for the UCF Sports. In order to

guarantee that the class-special dictionaries in LGSR are over-complete, for the KTH and UCF Sports random projection in dimension reduction [25] is adopted to reduce the dimension of sub-STV descriptor to 300 and 400, respectively.

In LGSR classification, the size of class-specific dictionaries is set to 150, 200, and 250 ($\mathbf{B}_k^1 = 150, \mathbf{B}_k^2 = 200, \mathbf{B}_k^3 = 250$)to the two datasets. Regularization parameters are set to $\lambda_1 = 1, \lambda_2 = 0.75, \lambda_1 = 0.6$ for the KTH, and $\lambda_1 = 1, \lambda_2 = 0.70, \lambda_1 = 0.55$ for the UCF Sports. Moreover, RE weights are set to $\beta_1 = 1, \beta_2 = 0.75, \beta_3 = 0.65$ for the KTH, and $\beta_1 = 1, \beta_2 = 0.80, \beta_3 = 0.55$ for the UCF Sports.

In experiment, leave-one-out cross-validation (LOOCV) strategy is used to evaluate the system performance. Specifically, for the KTH, in each LOO run, we use the videos of 24 subjects for training, and the videos of the remaining subject for test, and the recognition rate is the average value of the 25 runs. For the UCF sports, in each LOO, one video of each class is randomly selected as test data, the other videos are treated as training data, 100 LOO runs are carried out, and the recognition rate is the average value of the 100 runs.

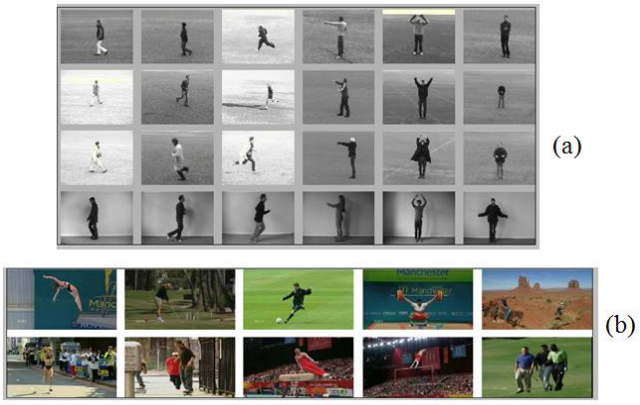


Figure 4. (A) Example Images from the KTH Dataset; (B) the UCF Sports Dataset

6.2. Human Action Datasets

The KTH dataset contains six classes of human action (i.e., boxing, hand clapping, hand waving, jogging, running, and walking). The actions are performed by 25 different subjects. Each subject performs four action videos in each class. Therefore, the KTH dataset includes $(25 \times 4 \times 6) = 600$ video clips with low-resolution $(160 \times 120$ pixels). Each action is performed in four scenarios: indoors, outdoors, outdoors with scale variation, and outdoors with different clothes. Examples of this datasets can be seen in Figure 4(a).

The UCF sports dataset includes 150 action videos, which are collected from various broadcast sports channel, such as BBC and ESPN. It contains 10 different actions: diving, golf swing, horse riding, kicking, lifting, running, skating, swing bar, swing floor, and walking. This dataset is challenging with a wide range of scenarios and viewpoints. Examples of this dataset can be seen in Figure 4(b).

6.3. Experimental Results and Discussion

To evaluate the recognition accuracy under the combination of different levels in HSS, the possible combinations of three levels of HSS are used, and the results are shown in Figure 5. It can be seen that only using the motion information from single level of HSS, the recognition rate is not satisfied; and the better performance is

obtained when the information from two levels is involved; when all motion information from three levels is used, the highest performance is achieved to the KTH and UCF Sports.

Table 1 shows the performance comparison between our system and some classical system published recently. The competing methods include local representation-based approaches [11-16], global representation-based approach. In detail, SC was used for feature coding together with BoF in [11], local feature distribution information was used in [14], ST context feature was employed in [12], sparse representation-based classification methods was applied in [13], the global representation method was adopted in [15], and a framework based on Elastic Manifold Embedding together with local interest point features to handle human action recognition in [16]. It demonstrates that our method achieves better performance than most of the competing methods.

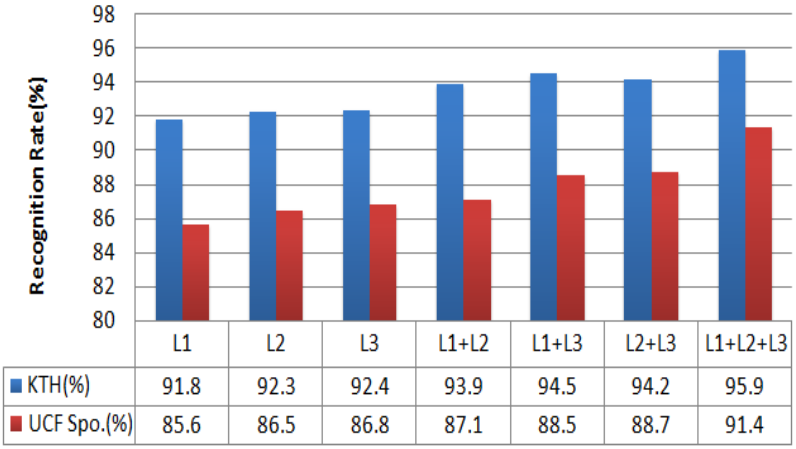


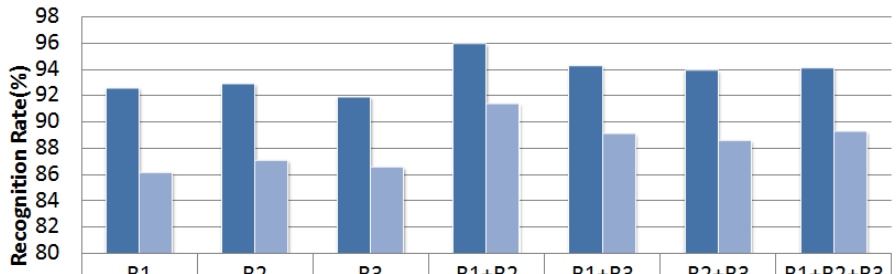
Figure 5. The Possible Combination Of Different Levels Of HSS Influence the Recognition Performance. L1 and L3 Denotes the Top Level and the Bottom Level of HSS. L1+ L2 Denotes the Combination of L1 and L2

Table 1. Our Recognition Performance Compares with the Classical Systems Published in the Past Years

Methods	Year	Action Model	KTH(%)	UCF Spo.(%)
Zhu *et al.* [11]	2010	SC was used for feature coding together with BoF	94.9	84.3
Wu *et al.* [12]	2011	Spatiot-temporal context feature was employed	94.5	91.3
Guha *et al.* [13]	2012	Sparse representation-based classification methods, and sub-STV model	—	91.1
Bregonzio *et al.* [14]	2012	Local feature distribution information was utilized	94.3	—
Saghafi *et al.* [15]	2012	The global representation method was adopted	92.6	—
Deng *et al.* [16]	2013	A framework based on Elastic Manifold Embedding together with local interest point features to handle human action recognition	96.9	88.4
Our system		HSS+MNF	95.9	91.4

Figure 6 shows the relationship between the MNF structure and the system performance. It can be found that the MNF with built by single neighborhood have weak discriminative power; when two neighborhoods are used to construct MNF, the better recognition accuracies are achieved, especially, the highest performance is reached when the MNFs consist of (R1+R2). However, the performance drops when three neighborhoods are involved in building MNFs. Such result can be explained that

when more neighborhoods are used to build MNFs, noisy data is entered into the MNF, and causes the weak discriminative power of MNF.



	R1	R2	R3	R1+R2	R1+R3	R2+R3	R1+R2+R3
KTH(%)	92.6	92.9	91.9	95.9	94.3	93.9	94.1
UCF(%)	86.2	87.1	86.6	91.4	89.1	88.6	89.3

Figure 6. The Relationship between the MNF Structure and the System Performance. R1, R2 and R3 Represent Level-1, Level-2, and Level-3 Neighborhood, Respectively, and their Nearest Neighbor Numbers are Set to 5, 7, and 9 (NB_1=5, NB_2=7, NB_3=9) for the KTH; and for the UCF Sports, (NB_1=3, NB_2=5, NB_3=7)

The confusion matrices for KTH and UCF sports datasets of our method are shown in Table 2.

Table 2. Confusion Matrixes on the Two Datasets. (A) Confusion Matrix on the KTH Dataset. S1 (Boxing), S2 (Hand-Waving), S3 (Hand-Clapping), S4 (Walking), S5 (Jogging), S6 (Running). (B) Confusion Matrix on the UCF Sports Dataset. S1 (Diving), S2 (Golfing), S3 (Kicking), S4 (Lifting), S5 (Horse-Riding), S6 (Running), S7 (Skating), S8 (Swing-Bench), S9 (Swing-High-Bar), S10 (Walking)

	s1(%)	s2(%)	s3(%)	s4(%)	s5(%)	s6(%)
s1(%)	**98.7**	0.85	0.45			
s2(%)	0.90	**98.3**	0.80			
s3(%)	0.80	0.70	**98.5**			
s4(%)				**93.1**	3.35	3.55
s5(%)				2.80	**93.2**	4.00
s6(%)				3.37	3.33	**93.3**

(a)

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10
s1(%)	**92.3**		2.85	2.55	2.30					
s2(%)		**91.6**		6.70	1.70					
s3(%)		3.53	**90.2**	3.61	3.66					
s4(%)	4.30		4.00	**91.7**						
s5(%)				5.30	**92.4**	1.30				
s6(%)				4.15	4.25	**91.6**				
s7(%)							**91.8**			9.20
s8(%)								**91.1**	8.90	
s9(%)								9.70	**90.3**	
s10(%)								8.60	2.70	**88.7**

(b)

7. Conclusion

To utilize the respond of STIP detector, which measures the strength of local motion changes in action video and is discarded in many cases, we propose to construct a multi-level HSS that provides different types of motion information about human action. Moreover, in order to overcome the drawbacks of the traditional neighborhood feature based on ε – neighborhood and KNN-neighborhood structure, a novel MNF feature is proposed, its main advantages is combining the similarity and position relationship between local features within a neighborhood. In classification, we revise the original LGSR for our classification task. The comparison with the recent systems

on the two public datasets validates the effectiveness of our HSS- and-PNF-based system.

Acknowledgements

This research is supported by National Nature Science Foundation of China (Grant no. 61271288), the National High Technology Research and Development Program (No.2012AA011503), the Research Fund for the Doctoral Program of Higher Education of China (No.20130185120014).

References

[1] J. Choi, W. Jeon and S.C. Lee, "Spatio-temporal pyramid matching for sports videos", ACM Multimedia, **(2008)**.
[2] I. Laptev, M. Marszalek, C. Schmid and B. Rozenfeld, "Learning realistic human actions from movies", IEEE International Conference on Computer Vision and Pattern Recognition (CVPR), **(2008)**, pp. 3222-3229.
[3] M. Marszalek, I. Laptev and C. Schmid, "Actions in context", Proceedings of International Conference on Computer Vision and Pattern Recognition (CVPR), **(2009)**, pp. 1006-1016.
[4] D. Han, L. Bo and C. Sminchisescu, "Selection and context for action recognition", IEEE International Conference on Computer Vision (ICCV), **(2009)**, pp. 1933-1940.
[5] J. Sun, X. Wu, S. Yan, L.F. Cheong, T.S. Chua and J. Li, "Hierarchical spatio-temporal context modeling for action recognition", Proceedings of CVPR, **(2009)**, pp. 2004-2011.
[6] A. Gilbert, J. Illingworth and R. Bowden, "Fast realistic multi-action recognition using mined dense spatio-temporal features", ICCV, **(2009)**, pp. 925-931.
[7] A. Fathi and G. Mori, "Action recognition by learning mid-level motion features", Proceedings of CVPR, **(2008)**, pp. 3063-3071.
[8] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang and Y. Gong, "Locality-constrained linear coding for image classification", Proceedings of CVPR, June **(2010)**, pp. 3360-3367.
[9] J. Yang, K. Yu, Y. Gong and T. Huang, "Linear spatial pyramid matching using sparse coding for image classification", Proceedings of CVPR, Miami, USA, **(2009)**, pp. 1794-1801.
[10] D. Xu, Y. Huang, Z. Zeng and X. Xu, "Human gait recognition using patch distribution feature and locality-constrained group sparse representation", IEEE Transactions on Image Processing, vol. 21, no. 1, **(2012)**, pp. 316-326.
[11] Y. Zhu, X. Zhao and Y. Fu, "Sparse coding on local spatial-temporal volumes for human action recognition", Proceedings of the Computer Vision (ACCV), Springer, Berlin, Germany, **(2010)**, pp. 660-671.
[12] X. Wu, D. Xu, L. Duan and J. Luo, "Action recognition using context and appearance distribution features", Proceedings of CVPR, June **(2011)**, pp. 489-496.
[13] T. Guha and R. K. Ward, "Learning sparse representations for human action recognition", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 34, no. 8, **(2012)**, pp. 1576-1588.
[14] M. Bregonzio, T. Xiang and S. Gong, "Fusing appearance and distribution information of interest points for action recognition", Pattern Recognition, vol.45, no.3, **(2012)**, pp.1220-1234.
[15] B. Saghafi and D. Rajan, "Human action recognition using Pose based discriminant embedding", Signal Processing, vol.27, no.1, **(2012)**, pp.96-111.
[16] [X. Deng, X. Liu and M. Song, "LF-EME: local features with elastic manifold embedding for human action recognition", Neurocomputing, vol. 99, no. 1, **(2013)**, pp.144-153.
[17] P. Dollar, V. Rabaud, G. Cottrell and S. Belongie, "Behavior recognition via sparse spatio-temporal features", Proceedings of the 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance, October, **(2005)**, pp. 65-72.
[18] J. C. Gemert, J. Geusebroek, C. J. Veenman and A. W. M. Smeulders, "Kernel codebooks for scene categorization", Proceedings of the 10th European Conference on Computer Vision (ECCV), Marseille, France, October **(2008)**, pp. 696-709.
[19] L. Liu, L. Wang and X. Liu, "In defense of soft-assignment coding", Proceedings of the 13th IEEE International Conference on Computer Vision (ICCV), Barcelona, Spain, November **(2011)**, pp. 2486-2493,
[20] T. Serre, L. Wolf and T. Poggio," Object recognition with features inspired by visual cortex", Proceedings of CVPR, **(2005)**, pp.304-311.
[21] M. Liu, S. Yan, Y. Fu and T. S. Huang, "Flexible X-Y patches for face recognition", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), April **(2008)**, pp. 2113-2116.
[22] C. P. Wei, Y. W. Chao and Y. R. Yeh, "Locality-sensitive dictionary learning for sparse representation based classification", Pattern Recognition, vol. 46, no. 5, **(2013)**, pp. 1277-1287.

[23] M. Bregonzio, T. Xiang and S. Gong, "Fusing appearance and distribution information of interest points for action recognition", Pattern Recognition, vol. 45, no. 3, **(2012)**, pp. 1220-1234.
[24] M. Liu, S. Yan, Y. Fu and T. S. Huang, "Flexible X-Y patches for face recognition", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), April **(2008)**, pp. 2113-2116.
[25] R. Baraniuk and M. Wakin, "Random projections on smooth manifolds", Foundations of computational mathematics, vol. 9, **(2004),** pp. 91-110.

Authors

Jiang-feng Yang, is a Ph.D student in the School of communication and information engineering, University of Electronic Science and Technology of China. He received his Master degree from Kunming University of Science and Technology in 2009. His current research interests include computer vision, human action recognition and motion detection.

Zheng Ma, is a professor in School of Communication and Information Engineering, University of Electronic Science and Technology of China. His current research interests include image processing, computer vision.