Image and Vision Computing 30 (2012) 227–235

Contents lists available at SciVerse ScienceDirect

Image and Vision Computing

journal homepage: www.elsevier.com/locate/imavis

Dynamic hand gesture recognition: An exemplar-based approach from motion divergence fields

Xiaohui Shen [a,*], Gang Hua [b], Lance Williams [c], Ying Wu [a]

[a] Northwestern University, Evanston, IL 60208, United States
[b] Stevens Institute of Technology, Hoboken, NJ 07030, United States
[c] Nokia Research Center Hollywood, Santa Monica, CA 90404, United States

ARTICLE INFO

Article history:
Received 29 June 2011
Received in revised form 1 November 2011
Accepted 4 November 2011

Keywords:
Hand gesture recognition
Divergence fields
Optical flow
Maximum Stable Extremal Regions
Term frequency-inverse document frequency (TF-IDF)

ABSTRACT

Exemplar-based approaches for dynamic hand gesture recognition usually require a large collection of gestures to achieve high-quality performance. Efficient visual representation of the motion patterns hence is very important to offer a scalable solution for gesture recognition when the databases are large. In this paper, we propose a new visual representation for hand motions based on the motion divergence fields, which can be normalized to gray-scale images. Salient regions such as Maximum Stable Extremal Regions (MSER) are then detected on the motion divergence maps. From each detected region, a local descriptor is extracted to capture local motion patterns. We further leverage indexing techniques from image search into gesture recognition. The extracted descriptors are indexed using a pre-trained vocabulary. A new gesture sample accordingly can be efficiently matched with database gestures through a *term frequency-inverse document frequency* (TF-IDF) weighting scheme. We have collected a hand gesture database with 10 categories and 1050 video samples for performance evaluation and further applications. The proposed method achieves higher recognition accuracy than other state-of-the-art motion and spatio-temporal features on this database. Besides, the average recognition time of our method for each gesture sequence is only 34.53 ms.

1. Introduction

Hand gestures are frequently used as intuitive and convenient ways of communications in our daily life, and the recognition of hand gestures can be widely applied in human computer interfaces, robot control, and augmented reality, *etc.*. Hand gestures can be conceptually divided into static gestures and dynamic gestures. Dynamic hand gestures usually provide a rich communication channel because of the incorporation of motion information, and are therefore more thoroughly investigated.

The approaches to dynamic hand gesture recognition can be categorized into model-based methods and exemplar-based methods. Model-based approaches include the Hidden Markov Model and its variants [1–5], Finite State Machines [6,7], dynamic Bayesian Networks [8], and topology-preserving self-organizing networks [9]. All these approaches assume that the hand has been detected and its articulated motion is tracked. Although they have delivered promising results, the robustness of these approaches is dependent on the prior success of (frequently challenging) hand detection and motion tracking. Furthermore, it is both data intensive and computationally difficult to train these models before they can be applied in recognition.

Various exemplar-based methods are therefore proposed to circumvent the difficulties of model learning, by leveraging invariant visual representations and direct matching of example gestures. Among those visual representations, local spatio-temporal features [10–12] are the most widely exploited, though most of them are used in human action recognition. Other descriptors include motion trajectories [13], spatio-temporal gradients [14] and global histograms of optical flow [15].

However, most of these methods try to directly match the exemplars, without offering a scalable solution for efficient matching when the exemplar database is large. A few others have adopted the bag-of-features framework with local spatio-temporal features for human action recognition [16,11]. Though they use a learned SVM classifier to perform recognition, it nevertheless could be incorporated with image indexing technique for scalable action recognition. However, those spatio-temporal features are not suitable in our hand gesture recognition scenario, as the durations of gestures are much shorter than human activities so that only limited spatio-temporal features can be extracted, and their effectiveness is accordingly degraded. Meanwhile, the extraction of these features is generally slow, though some efforts toward real-time extraction and recognition are being made [17,18]. Therefore, an efficient visual representation for real-time feature extraction and scalable hand gesture matching over large exemplar databases is still highly desirable.

* Corresponding author.
E-mail address: xsh835@eecs.northwestern.edu (X. Shen).

0262-8856/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2011.11.003

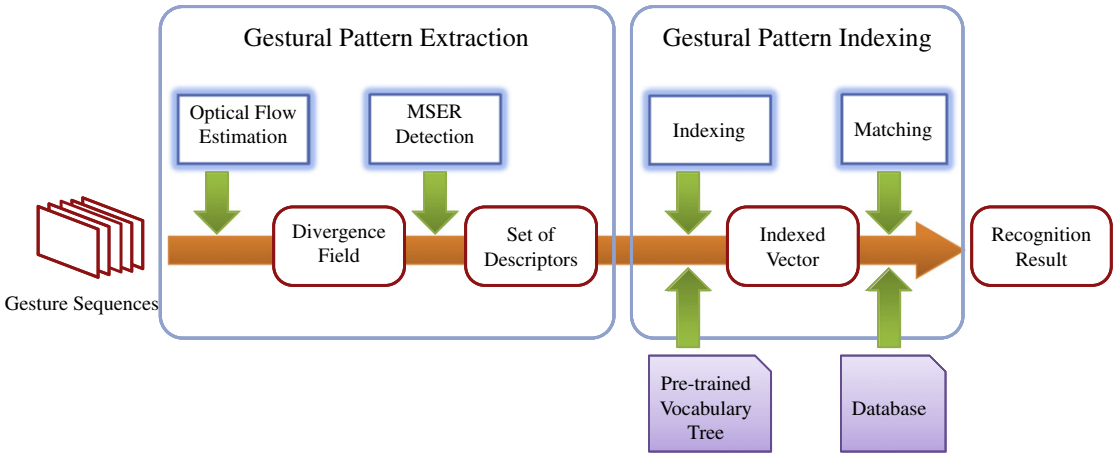


Fig. 1. The pipeline of the proposed method.

Toward this end, we propose a novel visual representation of dynamic hand gestures based on the divergence field of the hand flow motions. Given a gesture sequence, we extract the optical flow between any two consecutive frames. Their divergence fields are derived and normalized to gray-scale images, which transforms gestural motion patterns into spatial image patterns. Salient regions are then detected from the divergence field using a Maximally Stable Extremal Regions (MSER) [19,20] feature detector. A descriptor is subsequently extracted from each detected region to characterize the local motion patterns. The database gesture sequences with their descriptors are indexed by a pre-trained hierarchical vocabulary. A new gesture sequence is then recognized by matching against the database with a *term frequency-inverse document frequency* (TF-IDF) scheme [21], which is scalable to large databases. The pipeline of our method is illustrated in Fig. 1.

We have collected a sizable database of dynamic hand gestures with 10 categories and 1050 samples for evaluation, which will be shared with the research community for further study. In this database, gestures are performed on a two-dimensional static background, with a camera directly above the hand. Therefore the hand is the only moving object in the sequence. However, the background can be arbitrary, allowing background clutter. Meanwhile, the gestures can be performed with varying speed. The setup of the database simulates scenarios in which users make hand gestures in front of a camera sitting on a tabletop, which can be applied in human interactions with mobile devices.

Based on this setup, our method focuses on recognizing the gestural motions. Even if the gestures are performed with different poses, different speed, and on different background, as long as the motion patterns are the same, they are considered in the same category. We compared our method with state-of-the-art global motion descriptors and local spatio-temporal features on this database. Experiments show that the proposed approach outperforms both of them. The recognition rate of our method is 97.62%, with average recognition time 34.53 ms. In other words, our proposed approach presents not only a novel approach to motion pattern analysis, but also a scalable framework for dynamic hand gesture recognition over large databases.

This paper is a substantial extension of our conference paper [22]. Compared with [22], further details of our method are presented, and more extensive performance evaluation is conducted. We also give a more comprehensive literature review to introduce the background of our method and make the paper more self-contained. We further propose that our method can serve the purposes of gesture design. Therefore, this paper provides a more comprehensive and systematic report of our work. The rest of the paper is organized as follows: Section 2 introduces and discusses the related work. Section 3 presents our gestural pattern extraction approach based on motion divergence fields. Gesture indexing and matching is then discussed in Section 4. Section 5 gives extensive performance evaluation to validate the efficacy of the proposed method, and Section 6 draws the conclusion.

2. Related work

A comprehensive overview of recent gesture recognition methods can be found in [23]. Here we only introduce those model-based and exemplar-based methods that are closely related to our work, as well as the indexing techniques for scalable recognition.

The Hidden Markov Model (HMM) has been frequently used for gesture recognition. In [1], dynamic feature vectors are transformed to symbolic sequences by vector quantization, and subsequently modeled by a discrete HMM. Marcel et al. [2] train an Input–output HMM using EM, and apply it to recognize gestures from hand silhouettes extracted by segmentation and tracking. Some recent improvements over traditional HMM include the semantic network model (SNM) [3], the non-parametric HMM [4], and the Hidden Conditional Random Field [5]. These variants either reduce training efforts, or improve classification accuracy.

The Finite State Machine (FSM) is another popular model [6,7]. Davis and Shah [6] use a FSM to model four phases of a gesture, in which fingertips are detected to extract feature vectors. In [7], the states of the FSM are determined by the positions of detected head and hands. Besides these models, Suk et al. [8] propose to use dynamic Bayesian Networks to represent the relationship among gesture features based on motion tracking, and Flórez et al. [9] use the topology of a self-organizing neural network and its dynamics to determine hand postures and gestures.

Earlier exemplar-based methods use common motion features such as optical flows [14] and motion trajectories [13] as visual representations for gesture recognition. Kirishima et al. [24] extract Gaussian Density Features in regions surrounding selected interest points for learning and matching. Recently various spatio-temporal features and descriptors are proposed [10,11,12,25,26,27,15]. Laptev et al. [10] propose a method to detect Space-Time Interest Points (STIP) and adopt Histogram of Oriented Gradients (HOG) [28] and/or Histogram of Oriented Optical Flows (HOF) as descriptors, which achieves the state-of-the-art performance on action recognition [29]. Dollár et al. [10] extract the descriptors from space-time cuboids based on temporal Gabor filters. In [12], Hessian saliency measure for blob detection is extended to a spatio-temporal version. Space-time templates and shape features are also proposed in [26] and [25] respectively. Rodriguez et al. [27] generalize the Maximum Average Correlation Height (MACH) filter

to 3D spatio-temporal volumes. Chaudhryet al. [15] calculate a sequence of HOF and use Binet-Cauchy kernels on nonlinear dynamical systems.

However, most of these methods perform recognition by template matching or direct exemplar matching. Some of them [29] adopt bag-of-features representations [30], but still use SVM or nearest neighbor for classification, without using inverted files and indexing techniques to accelerate recognition procedures. Inverted index is the most popular data structure in document retrieval, and is further commonly used in large-scale image and object retrieval [21,31]. In our work, we leverage this data structure to our proposed motion patterns for hand gesture recognition over large databases.

There are also some methods devoted to speeding up video processing and feature extraction, by either designing fast online learning algorithms [32] or leveraging more powerful computing units [33] such as a GPU. Our method also achieves that goal by estimating optical flow on a GPU and fast feature extraction on the motion divergence fields.

3. Visual patterns of gestural motion

In this section, we present the proposed visual representation of gestural motions based on motion divergence fields. We first estimate the optical flow between every two consecutive frames in the gesture sequence. The divergence map of the optical flow field is then derived and normalized to a gray-scale image. MSER regions are then detected from the divergence map, and summary statistics are extracted as local motion descriptors.

3.1. The divergence field of optical flow

In a vector field, divergence is an operator that measures the magnitude of the source or sink of the field. Given a vector $\mathbf{F}=[F_1,F_2,\cdots,F_n]^T$ in a n-dimensional Euclidean space, the divergence of $\mathbf{F}$ can be calculated as:

$$\mathbf{divF}=\sum_{i=1}^{n}\frac{\partial F_i}{\partial x_i}, \tag{1}$$

where $[x_1,x_2,\cdots,x_n]^T$ are the Cartesian coordinates of the space where the vector field is defined.

Accordingly for an optical flow vector field $\mathbf{F}(x,y)=[u(x,y),v(x,y)]^T$, where $u(x,y)$ and $v(x,y)$ are respectively the horizontal and vertical components of optical flow at position (x,y), the divergence of $\mathbf{F}$ is:

$$\mathbf{divF}=\frac{\partial u}{\partial x}+\frac{\partial v}{\partial y}. \tag{2}$$

Fig. 2 presents an example of transforming a flow motion field into a divergence field. Fig. 2(a) is the first frame of an image pair, and Fig. 2(b) and (c) are the visualizations of u and v respectively. The corresponding divergence field after normalized to gray-scale is shown in Fig. 2(d). We calculate the optical flow using the Lucas–Kanade algorithm [34] and implement the algorithm on the GPU to speed up the processing frame rate. Lucas–Kanade optical flow estimation, as applied here, relies on local contrast and texture, and is valid only for small motions. Since no multi-resolution estimation is applied, the computed flow is not valid for large areas with little texture, such as the interior of the hand region in the example. As a result, the flows in these areas can hardly be estimated. However, such estimation of optical flow proved a sound basis for discrimination in our experiments, even if the flow values are not absolutely accurate. As we can see, the divergence field of optical flow has filtered out most flow noise in the background and provides a clear shape of the hand, which ensures that most MSER regions are located on the hand, which we will discuss in the next section. We proceed to present the extraction of the local motion descriptors.

3.2. Local descriptor extraction

Once we obtain the divergence field of optical flow, Maximally Stable Extremal Regions (MSER) [19] are detected from the map. The MSER region is defined exclusively by an extremal property of the intensity function in the region and on its outer boundary, and therefore has many useful properties, including invariance to affine transformation of image intensities, stability, and allowing multi-scale detection. MSER has been widely used in image matching and object recognition, and has led to better recognition performance [20] in several applications. In our proposed framework, each MSER region is fitted by an ellipse, as shown in Fig. 2(e).

Because the background is static, most MSER regions detected in the motion divergence field are on the boundary of the hand or within the hand. The features extracted from these regions therefore are not

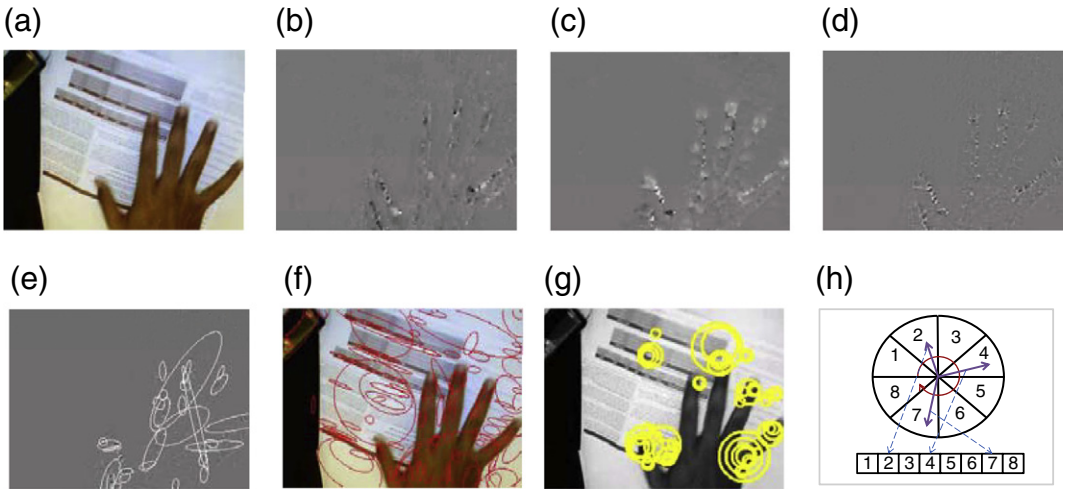


Fig. 2. Gestural pattern extraction.(a) the first frame of an image pair, (b) the u component of the estimated optical flow, (c) the v component of the estimated flow, (d) the divergence field, (e) MSER detection on the divergence field in (d), (f) MSER detection directly on the image in (a), (g) the detected STIP in the same frame, (h) calculating a histogram of optical flow orientations with 8 bins from MSER regions.

mixed with background clutter. As a comparison, Fig. 2(f) shows the MSER regions detected directly from the image in Fig. 2(a), which are as frequently detected in background regions as within the moving hand. We also performed space-time interest points (STIP) detection on the same frames, which is a state-of-the-art spatio-temporal feature. The results are shown in Fig. 2(g). As we can see, STIP detection is also distracted by background texture, which makes their detectors less discriminative on hand motion.

In each detected MSER region, we calculate a histogram of the orientations of the optical flow vectors. The orientations of optical flow can be calculated from $u(x,y)$ and $v(x,y)$ and have a range of $[0,2\pi]$. All the orientations are then bi-linearly quantized and aggregated into discrete bins with their magnitudes as weights. Fig. 2(h) provides a simple illustration, in which the histogram has 8 bins. In practice we set the bin number to be 80. The histogram is finally normalized to have unit L1-norm.

By choosing histograms of flow orientations as our local descriptors, we get rid of the magnitude of optical flows. That is because, the speed of gestures varies widely, particularly among different users. A good gesture recognition algorithm therefore should be relatively insensitive to the speed with which a gesture is performed. This suggests orientations of hand movement as significant measures for recognition. It is also validated by other action recognition methods using histograms of flow orientations [29,15]. What we are seeking here is a set of discriminative descriptors for each distinct gesture. We validate in our experiments that such descriptors are already highly discriminative, irrespective of the fact that we adopted a simple algorithm to estimate optical flow.

After local descriptor extraction, each divergence field is represented by a set of local descriptors, and a hand gesture is a sequence of such descriptor sets. By such visual representation of the motion patterns, we dispense with relatively complicated motion estimation techniques such as segmentation and tracking. Meanwhile, MSER detection, with linear time implementation [35], can be performed at modest computational expense. As a result, the whole feature extraction process in our method is very efficient.

4. Motion gestural pattern indexing

Once we get the local descriptors for a gesture sequence, each descriptor is quantized by a pre-trained vocabulary, and indexed using inverted files. We then match a test gesture sequence with the database using a *term frequency-inverse document frequency* (TF-IDF) weighting scheme. This technique is widely adopted in large-scale visual search. We extend it to gestural motion pattern matching and offer a scalable solution for gesture recognition.

4.1. Building a vocabulary

Before gesture indexing and matching, we need to build a vocabulary for the descriptors first. In the context of image search and object recognition, a vocabulary is a structure of cluster centers of a set of training descriptors. Each cluster center is called a visual word. The extracted features in the image are then quantized through the vocabulary and assigned to their closest visual words for subsequent matching. There are several methods to build the vocabulary [21,31]. Here we use hierarchical k-means clustering (HKM) [21] to build a vocabulary tree, which is a hierarchical structure of visual words. Fig. 3 shows a simple vocabulary tree with 6 branches and 3 levels. Each node in that tree represents a visual word.

Hierarchical k-means clustering is performed as follows. Given the training dataset, a k-means clustering process is first performed to determine k cluster centers, where k is the branch factor of the tree, i.e., the number of children of each node. In Fig. 3, $k=6$. These k centers represent the nodes in the first level of the tree. The training descriptors are then branched to k groups according to their distances from the cluster centers. In each group, k-means clustering is further performed to define

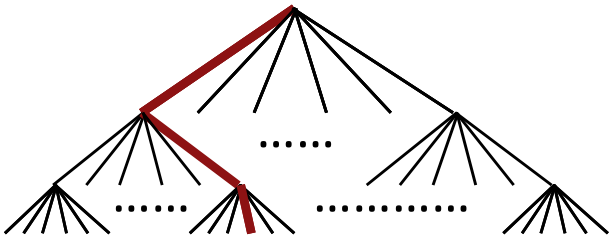


Fig. 3. A vocabulary tree with 6 branches and 3 levels.

k new cluster centers, which are then the children of the original center. The same process is carried out recursively until the tree achieves a predefined maximum depth. For a vocabulary tree with k branches and l levels, the total number of leaf nodes would be k^l.

Once the vocabulary tree is built, the descriptors in an image can be quantized by comparing with the descriptors of the k nodes at each level, and associated with the closest one. Each descriptor thus has a path from the root to a leaf in the tree. The red line in Fig. 3, for example, is a path for one descriptor. Such a path can be encoded by a single integer at the leaf, and used for indexing and matching, as described in the following section. Therefore, after quantization, each descriptor can be represented by a single integer, which significantly reduce the memory cost, and make the subsequent matching more efficient as well. Note that each quantization process for a descriptor involves only $k*l$ comparisons. The computational cost is logarithmic in the number of leaf nodes, which is the principal advantage offered by hierarchical structure.

4.2. Indexing a single image

After all the descriptors of a query image are quantized through the vocabulary tree, the image can be matched with the database images by comparing the similarities of the paths of their descriptors. Consider that n_i and m_i are the number of descriptors quantized to the i-th node (visual word) in the query image and in a database image respectively, the distance between the query image and the database image can be defined as:

$$\begin{aligned} &a_i = n_i w_i, \qquad b_i = m_i w_i \\ &d(\mathbf{a},\mathbf{b}) = \left\| \frac{\mathbf{a}}{\|\mathbf{a}\|} - \frac{\mathbf{b}}{\|\mathbf{b}\|} \right\|_p^p = \sum_i |a_i - b_i|^p \end{aligned} \tag{3}$$

where w_i is the weight of the i-th node in the vocabulary tree; p indicates Lp-norm. We use L1-norm in our experiments. The weight w_i can be defined based on entropy:

$$w_i = \log \frac{N}{N_i} \tag{4}$$

where N is the number of images in the database, and N_i is the number of images that have descriptors quantized to the i-th node, which in text analysis is called *inverse document frequency*. It is found in practice that leaf nodes contain the most information, and sometimes only the leaf nodes are used in image matching for convenience.

Usually there are thousands of leaf nodes in a vocabulary tree, with only hundreds or dozens of descriptors in an image. As a result, **a** and **b** are both sparse vectors. After we normalize **a** and **b** to have unit magnitude, the distance of **a** and **b** in Eq. (3) can be further rewritten as:

$$\begin{aligned} d(\mathbf{a},\mathbf{b}) &= \textstyle\sum_i |a_i - b_i|^p = \sum_{b_i=0} |a_i|^p + \sum_{a_i=0} |b_i|^p + \sum_{a_i\neq 0, b_i\neq 0} |a_i - b_i|^p \\ &= \textstyle\sum_i |a_i|^p + \sum_i |b_i|^p - \sum_{a_i\neq 0, b_i\neq 0} \left(|a_i - b_i|^p + |a_i|^p + |b_i|^p\right) \\ &= 2 - \textstyle\sum_{a_i\neq 0, b_i\neq 0} \left(|a_i - b_i|^p + |a_i|^p + |b_i|^p\right) \end{aligned} \tag{5}$$

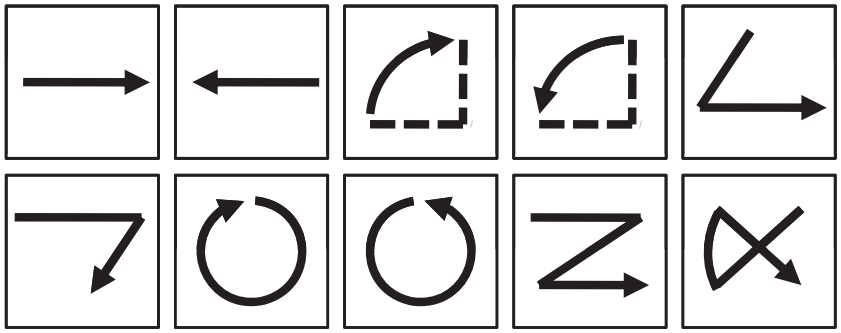


Fig. 4. Ten dynamic gestures: move right, move left, rotate up, rotate down, move down-right, move right-down, clockwise circle, counterclockwise circle, "Z", and "cross".

Only the distances between the non-zero elements of the vectors are calculated. This allows us to use inverted files to avoid direct matching of the query vector with all the image vectors in the database. An inverted file for each node is a file recording the number of images that have at least one descriptor quantized to that node, and the IDs of these images, along with the number of descriptors in these images that are quantized to that node, which in text analysis is called *term frequency*. Our distance measure incorporates a *term frequency-inverse document frequency* (TF-IDF) weighting scheme. When the descriptors of the query image are all quantized, only the inverted files of the nodes corresponding to the non-zero elements of the query vector are looked up. The distances of the query image to each of the images recorded in the inverted files can be gradually accumulated using Eq. (5). By using inverted files for matching in the TF-IDF scheme, the computational cost of image matching is significantly reduced. This approach enables efficient search even if there are millions of leaf nodes and database images.

4.3. Indexing a gestural motion sequence

The image indexing technique introduced in Section 4.2 only works in single image search, while in Section 3 a hand gesture is converted to a variable-length sequence of divergence images. In this section, we will extend image indexing for gesture sequence matching.

One straightforward solution is to uniformly sample frames from a gesture sequence. Using the method in Section 4.2, each sampled frame can be indexed to form a vector $\mathbf{a}^i = [a_1^i, a_2^i, \cdots, a_M^i]^T$, where i indicates the i-th sampled frame, and M is the number of leaf nodes in the vocabulary tree. We concatenate the vectors of all the sampled frames to form a new vector, which represents the indexing results of the entire gesture sequence. The distance of two gesture sequences accordingly can be calculated as:

$$d(\mathbf{a},\mathbf{b}) = \sum_{i=1}^{n} d\left(\mathbf{a}^i,\mathbf{b}^i\right) = \sum_{i=1}^{n}\sum_{j}\left|a_j^i - b_j^i\right|^p \tag{6}$$

where n is the total number of sampled frames.

This extension, though straightforward and simple, is very important in our method for the following two reasons:

- It normalizes the gestures to vectors with the same length, which removes the factor of gesture duration, and enables recognition of gestures with substantial variation in speed;
- The direction of optical flow changes continually in some gestures (e.g., drawing a circle). Experimental results have validated that by sampling above a critical rate, the dynamic changes of the motion patterns can be successfully preserved in the concatenated vectors.

In this matching scheme, the spatial information of the descriptors is discarded in the quantization and indexing step. However, such information is usually quite useful. Therefore, a post-verification step incorporating the spatial information is used to re-rank those top returned candidates. This is a common step in image search and object recognition [31].

We retain the top k candidate gestures after the indexing and matching procedure. In the post-verification, we first estimate the geometric center of the hand for each sampled frame according to the positions of the detected MSER regions. The centers in the query and those in the candidates are then compared to assign a score to each candidate. If the centers in a database sequence are closer to the centers in the query, its corresponding score would be high, and vice versa. The aggregated scores for each gesture category can then be obtained from these top k scores, and the query gesture is assigned to the category with the highest score.

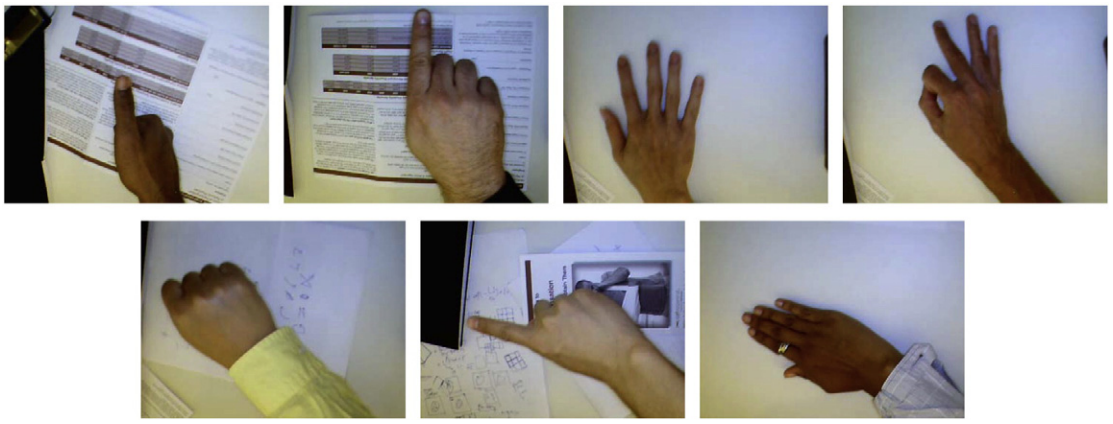


Fig. 5. Seven postures: thumb, index finger, hand - fingers extended, "OK"(thumb and forefinger loop), fist, index finger with 90° rotation and hand with 90° rotation.

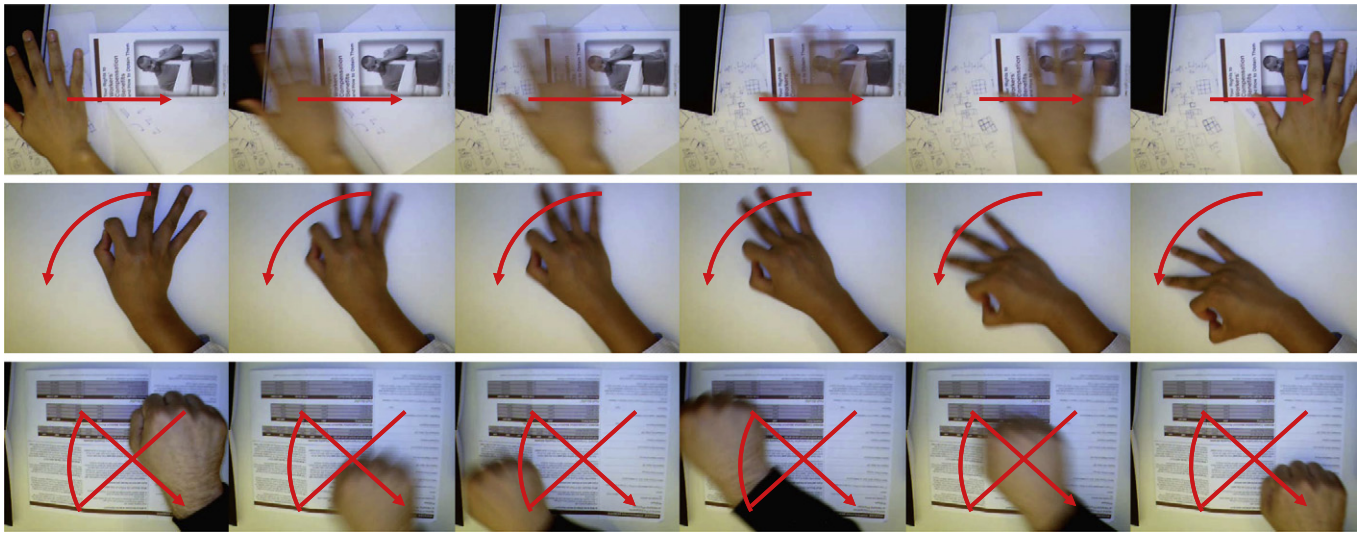


Fig. 6. An example sequence in the collected database. The database has 10 categories of gestures and 1050 samples in total.

5. Experiments

5.1. The database

The database contains 10 categories of dynamic hand gestures in total: move right, move left, rotate up, rotate down, move down-right, move right-down, clockwise circle, counterclockwise circle, "Z", and "cross", as shown in Fig. 4. In the collection process, each person is asked to perform these ten actions with seven postures as illustrated in Fig. 5: thumb, index finger, hand - fingers extended, "okay"(thumb and forefinger loop), fist, index finger with 90° rotation and hand with extended fingers at 90° rotation. Each subject contributes 70 gesture samples to our database. We collected 1050 sample gestures performed by 15 subjects. Fig. 6 provides some example sequences. As we can see, the background as well as the skin colors of the hands are very diverse, and the captured sequences contain severe motion blurs. Both conditions are common in real applications. We consider this a representative database that is useful not only for dynamic hand gesture recognition but also for static hand pose estimation as well.

Since our method addresses dynamic hand gesture recognition, we focus on recognizing the 10 dynamic gestures in our experiments. Thus the samples with the same action in different hand postures are considered as one category, and each category accordingly has 105 samples.

The evaluation is performed in a user-independent way, as leave-one-subject-out cross-validation is used. That is, given a test gesture sequence, the gestures performed by the same subject are excluded from the database. The test gesture is then matched with the remaining database gestures, and its category is collectively determined by its k-nearest neighbors after indexing, matching and post-verification, as introduced in Section 4.3.

5.2. Determining the parameters

There are two main parameters in our framework to be determined: the size of the vocabulary tree (the number of leaf nodes), and the number of sampled frames in a gesture sequence for indexing, as described in Section 4.3.

We have collected 2257851 descriptors in total, and chose different branch factors and levels to build the vocabulary tree. The recognition performance with different tree sizes is shown in Fig. 7. Contradicting the observation in [21] that a larger vocabulary tree would improve performance, in our experiments the recognition rate is already very high with only 512 leaf nodes. The performance remains stable when the size is smaller than 2000, and drops slightly when the size is larger. However, the recognition rate is still above 96% when the size is over 10,000. The vocabulary tree with 9 branches and 3 levels (729 leaf nodes) achieves the highest recognition rate, which is 97.62%. This tree will be used in the following experiments.

We tried encoding sequences in different numbers of sampled frames, and depict the results in Fig. 8. As shown by the red line in Fig. 8, recognition performance is quite robust at different frame sampling rates. Even if only 3 frames are sampled, the recognition rate has already achieved nearly 96%. When the frame sampling rate is larger than 7, the performance cannot be further improved. That is probably because, for discriminating gestures in our database, dynamic

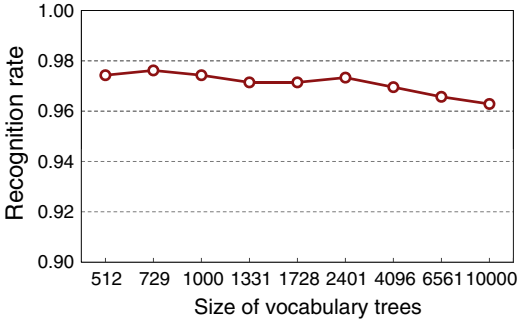


Fig. 7. Recognition performance with different sizes of vocabulary tree.

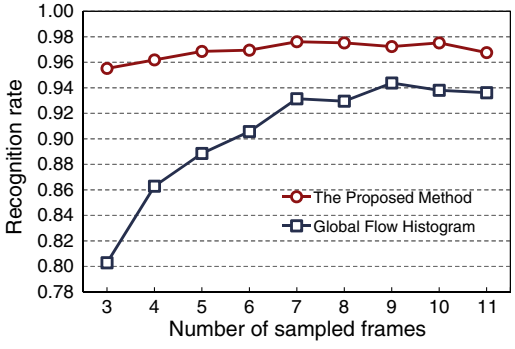


Fig. 8. Recognition performance with different sampling rates.

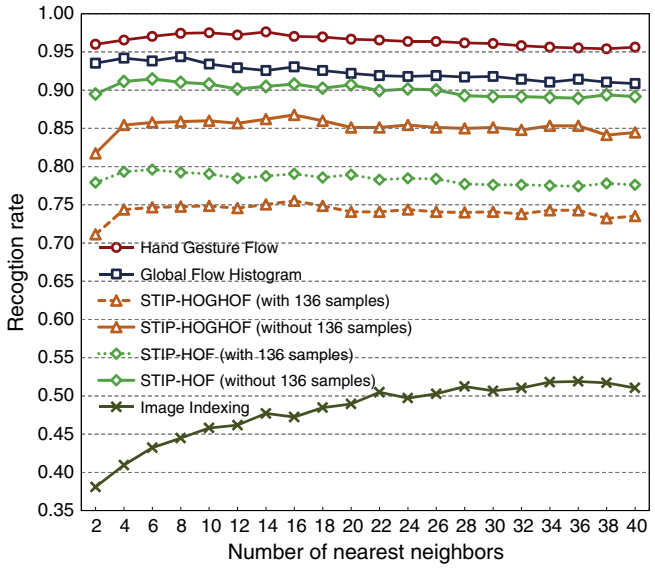
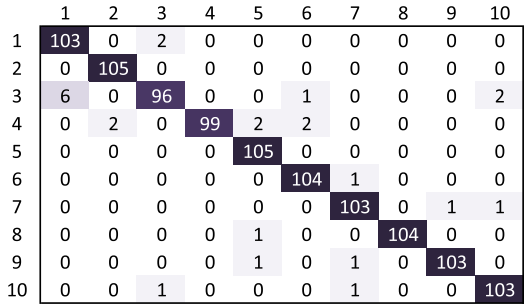


Fig. 9. Performance evaluation with different top k candidates. Our method consistently outperforms others.



	1	2	3	4	5	6	7	8	9	10
1	103	0	2	0	0	0	0	0	0	0
2	0	105	0	0	0	0	0	0	0	0
3	6	0	96	0	0	1	0	0	0	2
4	0	2	0	99	2	2	0	0	0	0
5	0	0	0	0	105	0	0	0	0	0
6	0	0	0	0	0	104	1	0	0	0
7	0	0	0	0	0	0	103	0	1	1
8	0	0	0	0	1	0	0	104	0	0
9	0	0	0	0	1	0	1	0	103	0
10	0	0	1	0	0	0	1	0	0	103

Fig. 11. Confusion matrix of our recognition result. The order of gesture categories: move right, move left, rotate up, rotate down, move down-right, move right-down, clockwise circle, counterclockwise circle, "Z", and "cross". Each category contains 105 samples. Most misclassification are in *Rotate Up* and *Rotate Down*.

information of the motion patterns has already been fully captured in 7 frames.

5.3. Comparisons

We compared our method with three other methods:

5.3.1. Image indexing

This is a baseline method in which MSER detection is directly performed on the image sequences, without optical flow estimation. A HOG descriptor is then extracted from each detected MSER region. The image descriptors are then sampled and indexed using the method in Section 4. Compared with our method which indexes and matches motion patterns from the divergence field, this baseline method directly matches appearance patterns of the hand. Variations in appearance would seem to necessitate a very large training set, and matching is greatly influenced by features extracted from the background. The performance of this method is poor in our dataset, barely above 50%, as shown in Figs. 9 and 10(a).

5.3.2. Global histograms of oriented optical flow

The second method is one adapted from Chaudhry et al. [15]. In their original paper, they use a global histogram for the entire oriented optical flow field and then extract the dynamics of the histograms for periodic action recognition. However, since only the dynamics (changes) in the histograms are used, this approach cannot be expected to discriminate some gestures in our database (e.g. constant motions left or right, in which the histograms do not necessarily change). To make their method work in our scenarios for a fair comparison, once we extract a sequence of global histograms, we also resample frames to normalize sequence length, as proposed in Section 4.3, and calculate the χ^2 distance for the concatenated histograms:

$$d(\mathbf{a},\mathbf{b}) = \frac{1}{2}\sum_i \frac{|a_i - b_i|^2}{a_i + b_i}. \tag{7}$$

The recognition result is then determined by a k-Nearest Neighbor classifier (k-NN). The best recognition rate of this method is 94.38%, which validates the discriminative power of histograms of oriented optical flow. However, our method is more robust than the global histogram. Fig. 9 shows the recognition results with different numbers of top k examples for classification. The recognition rate of our method is always higher than the global histogram. Moreover, the global histogram method is more sensitive to the sampling rate. As shown in Fig. 8, the classification performance of global histograms drops dramatically when the sampling rate is under 7. This strongly suggests that the local descriptors for motion patterns in our method are more robust than global flow histograms in capturing and discriminating the dynamics of hand gestures.

On the other hand, benefiting from the indexing scheme, the average recognition time of our method is 34.53 ms, which is only about 20% of the recognition time of global histogram matching, as shown in Fig. 10(b). The recognition time of histogram matching is linear in the number of gestures in the database. Given a database with more than 10 thousand gestures, real-time recognition by nearest-neighbor classification is not currently feasible, while our method is readily scalable to large databases. Direct image indexing also takes more recognition time than our method, which is probably because

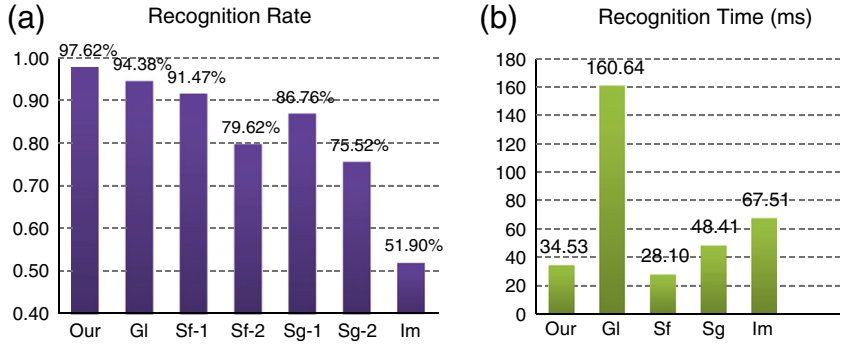


Fig. 10. Comparisons of all the methods on recognition performance and time. Our: the proposed method, Gl: global flow histogram, Sf: STIP-HOF, (1) without 136 samples, (2) with 136 samples, Sg: STIP-HOG + HOF, (1) without 136 samples, (2) with 136 samples, Im: image indexing. Our method achieves the best performance with very high speed.

many more MSER regions are detected directly from the image sequences than from their optical flow divergence fields.

5.3.3. Space-time interest points (STIP)

STIP with HOG/HOF descriptors [10,16] is a representative local spatio-temporal feature which achieves the state-of-the-art performance in several action datasets [29]. We use their provided code [36] to detect STIP and extract two types of descriptors in each detected STIP: HOG + HOF, and HOF. Both of them are evaluated for comparison. In their original paper, after the descriptors are quantized through the vocabulary, a SVM is trained for classification. To make the comparison fair, we adopt the indexing framework as introduced in Section 4, which is the same as in our method, for gesture matching. Since their features already consider the gesture sequences as spatio-temporal volumes, no sampling is used for their method.

Unlike other human activities, hand gestures usually last a short period of time. Some simple gestures, such as move left or right, can even be finished within one second. This distinctive property of hand gestures has largely lessened the efficacy of spatio-temporal features. In our evaluation, only a limited number of STIP are detected from some gesture sequences with short durations. Though we have tuned the parameters trying to detect as many points as possible, there are still 136 out of 1050 sequences that do not contain any detected STIP, which therefore cannot be recognized.

If we consider these 136 samples as failure cases, then the best recognition rates achieved are 75.52% with HOG + HOF descriptors and 79.62% with HOF descriptors respectively. If we exclude all these 136 samples without STIP, and evaluate their methods on the remaining 914 samples, then their best recognition rates are 86.76% with HOG + HOF and 91.47% with HOF. The results are also shown in Figs. 9 and 10(a). Using HOF as descriptors achieves better performance than combining HOG and HOF. That is because the appearance information captured by HOG may reduce the distances between the gesture sample to be recognized and those database samples with the same pose but different motions, and accordingly weakens the power of the descriptors in motion recognition.

We can observe from Fig. 9 that even if we exclude these 136 samples in the evaluation of their methods, their recognition rates are still lower than ours. Despite the discriminative power of the features, their feature points are distracted by background clutter, while in our method the detected MSER regions are mainly located on the moving hands, as illustrated in Fig. 2.

Since we adopted the same indexing framework for gesture matching, the recognition time of using STIP is fast, as shown in Fig. 10(b). However, the feature extraction procedure of STIP is far from real-time. The average speed of extracting STIP on 320×240 video sequences is ~2 *fps*, while our method can achieve ~30 *fps*. Considering the performance and the processing time, our method is more suitable for recognition in this hand gesture dataset.

5.4. Failure cases and gesture design

Fig. 11 shows the confusion matrix of the recognition results of our method. We can see that most of the misclassifications exist in *Rotate Up* and *Rotate down*. Some misclassifications are due to the similarities of these gestures with others such as *Move Right* and *Move left*, as well as the ambiguity when users perform these gestures. For example, in Fig. 12, the user is doing the *Rotate Up* gesture. However, the rotation of the hand is not sufficient, and our method misclassifies the gesture as *Move Right*.

These gesture recognition results, from another perspective, could provide guidance on hand gesture design. Given an efficient hand gesture recognition method, by analyzing the misclassification samples, we can find out which gesture is easily confused with others, such as *Rotate Up* vs. *Move Right*, and *Rotate Down* vs. *Move Left*. These gestures then can be removed while more distinctive gestures are retained. In application we need to develop a gesture recognition system with very high accuracy, not only by using an efficient recognition method but also by selecting discriminative gestures, and our method can well serve the purpose of designing gestures.

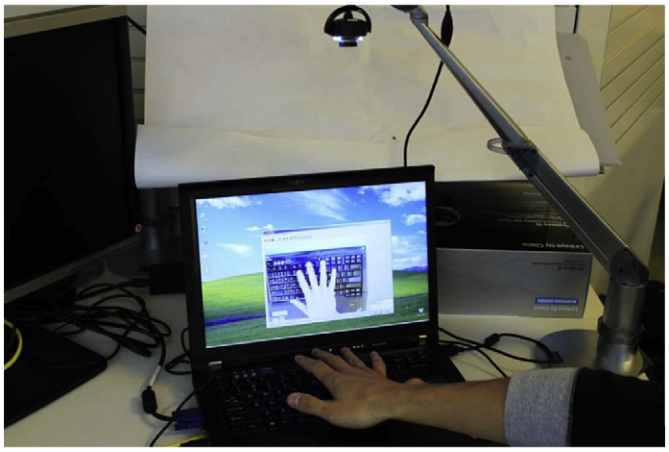


Fig. 13. A snapshot of our prototype system.

5.5. Applications

We built a prototype system and applied our method for live hand gesture recognition. Fig. 13 shows a snapshot of this system. Gesture extraction, i.e., automatic detection of the start and end of the gestures, is implemented in the system by a global motion threshold. The system can recognize live hand gestures in real-time (~30 *fps*) for 320×240 video sequences, with high recognition quality.

6. Conclusions

In this paper we present a new visual presentation for hand gesture motions. By calculating the divergence fields of optical flow, a gestural motion sequence is converted to an image sequence encoding the motion information. It allows us to use modern image feature detectors such as MSER to extract the salient regions and transform motion patterns into discriminative spatial image patterns. We also offer a scalable solution by extending the image indexing prototype to hand gesture recognition. The proposed framework achieves high recognition accuracy in our database, while scalable to larger databases.

Fig. 12. An example in which our method fails to recognize the gesture. In a *Rotate Up* gesture, the rotation of the hand is not great enough to be detected, and our method misclassifies the gesture as *Move Right*.

In our future work, we will design more meaningful gestures and apply our method to larger-scale databases. We also believe that the proposed approach is applicable to more general action/activity recognition tasks.

References

[1] J. Yamato, J. Ohya, K. Ishii, Recognizing human action in time-sequential images using hidden markov model, CVPR, 1992.
[2] S. Marcel, O. Bernier, D. Collobert, Hand gesture recognition using input–output hidden Markov models, FG, 2000.
[3] S. Rajko, G. Qian, T. Ingalls, J. James, Real-time gesture recognition with minimal training requirements and on-line learning, CVPR, , 2007.
[4] A. Elgammal, V. Shet, Y. Yacoob, L.S. Davis, Learning dynamics for exemplar-based gesture recognition, CVPR, 2003.
[5] S. Wang, A. Quattoni, L.P. Morency, D. Demirdjian, T. Darrell, Hidden conditional random fields for gesture recognition, CVPR, 2006.
[6] J. Davis, M. Shah, ECCV, Recognizing Hand Gestures, , 1994.
[7] P. Hong, M. Turk, T.S. Huang, Gesture modeling and recognition using finite state machines, FG, 2000, pp. 410–415.
[8] H. Suk, B. Sin, S. Lee, Recognizing hand gestures using dynamic bayesian network, FG, 2008.
[9] F. Flórez, J.M. García, J. García, A. Hernández, Hand gesture recognition following the dynamics of a topology-preserving network, FG, 2002.
[10] I. Laptev, T. Lindeberg, Space-time interest points, ICCV, 2003.
[11] P. Dollár, V. Rabaud, G. Cottrell, S. Belongie, Behavior recognition via sparse spatio-temporal features, ICCV VS-PETS, 2005.
[12] G. Willems, T. Tuytelaars, L.J.V. Gool, An efficient dense and scale-invariant spatio-temporal interest point detector, ECCV, 2008.
[13] M.H. Yang, N. Ahuja, M. Tabb, Extraction of 2d motion trajectories and its application to hand gesture recognition, IEEE Trans. on PAMI 24 (2002) 1061–1074.
[14] W.T. Freeman, M. Roth, Orientation histograms for hand gesture recognition, FG, 1995.
[15] R. Chaudhry, A. Ravichandran, G. Hager, R. Vidal, Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions, CVPR, 2009, pp. 1932–1939.
[16] I. Laptev, M. Marszałek, C. Schmid, B. Rozenfeld, Learning realistic human actions from movies, CVPR, 2008.
[17] A. Patron-Perez, I. Reid, A probabilistic framework for recognizing similar actions using spatio-temporal features, BMVC, 2007.
[18] T.-H. Yu, T.-K. Kim, R. Cipolla, Real-time action recognition by spatiotemporal semantic and structural forests, BMVC, 2010.
[19] J. Matas, O. Chum, M. Urban, T. Pajdla, Robust wide baseline stereo from maximally stable extremal regions, BMVC, 2002.
[20] P. Forssen, D. Lowe, Shape descriptors for maximally stable extremal regions, ICCV, 2007.
[21] D. Nistér, H. Stewénius, Scalable recognition with a vocabulary tree, CVPR, 2006, pp. 2161–2168.
[22] X. Shen, G. Hua, L. Williams, Y. Wu, Motion divergence fields for dynamic hand gesture recognition, FG, 2011.
[23] S. Mitra, T. Acharya, Gesture recognition: a survey, IEEE Trans. on Systems, Man and Cybernetics — Part C 37 (3) (2007) 311–324.
[24] T. Kirishima, K. Sato, K. Chihara, Real-time gesture recognition by learning and selective control of visual interest points, IEEE Trans. on PAMI 27 (2005) 351–364.
[25] L. Gorelick, M. Blank, E. Shechtman, M. Irani, R. Basri, Actions as space-time shapes, PAMI 29 (12) (2007) 2247–2253.
[26] A. Efros, A. Berg, G. Mori, J. Malik, Recognizing Action at a Distance, 2003, pp. 726–733.
[27] M.D. Rodriguez, J. Ahmed, M. Shah, Action mach: A spatio-temporal maximum average correlation height filter for action recognition, CVPR, 2008.
[28] N. Dalal, B. Triggs, Histograms of oriented gradients for human detection, CVPR, 2005.
[29] H. Wang, M.M. Ullah, A. Kläser, I. Laptev, C. Schmid, Evaluation of local spatio-temporal features for action recognition, BMVC, , 2009.
[30] L. Fei-Fei, P. Perona, A Bayesian Hierarchical Model for Learning Natural Scene Categories, , 2005.
[31] J. Philbin, O. Chum, M. Isard, J. Sivic, A. Zisserman, Object retrieval with large vocabularies and fast spatial matching, CVPR, 2007.
[32] R. Lockton, A.W. Fitzgibbon, Real-time gesture recognition using deterministic boosting, BMVC, 2002, pp. 817–826.
[33] F.S. Chen, C.M. Fu, C.L. Huang, Hand gesture recognition using a real-time tracking method and hidden Markov models, Image and Vision Computing 21 (2003) 745–758.
[34] B. Lucas, T. Kanade, An iterative image registration technique with an application to stereo vision, IJCAI, 1981, pp. 674–679.
[35] D. Nistér, H. Stewénius, Linear time maximally stable extremal regions, ECCV, 2008, pp. 183–196.
[36] http://www.irisa.fr/vista/Equipe/People/Laptev/download.html.