FIRST-PERSON ACTION RECOGNITION THROUGH VISUAL RHYTHM TEXTURE DESCRIPTION

Thierry Pinheiro Moreira[1], *David Menotti*[2], *Helio Pedrini*[1]

[1] Institute of Computing - University of Campinas
Campinas, SP, Brazil, 13083-852

[2] Department of Informatics - Federal University of Paraná
Curitiba, PR, Brazil, 81531-990

ABSTRACT

First-person action recognition is a recent problem in computer vision, where an observer wears body cameras to understand and recognize actions from the captured video sequences. Technological advances have made it possible to offer small wearable cameras that can be attached onto bike helmets, belts, animal halters, among other accessories. Examples of potential applications include sports, security, healthcare, visual lifelogging, among others. In this paper, we propose a novel approach to first-person action recognition that consists in encoding video appearance, shape and motion information as visual rhythms and describing them through texture analysis. Experiments are conducted on the DogCentric Activity and JPL First-Person Interaction datasets, showing accuracy improvement over the baselines.

Index Terms— Action recognition, first person, visual rhythms, video analysis, texture description

1. INTRODUCTION

First-person activity recognition [1, 2, 3] is a growing area of research due to the appearance of a new category of devices: wearable gadgets. People are able to make egocentric videos. Users record videos playing sports (such as surfing, pakour, football, and climbing), doing everyday activities and working.

Users frequently interact with portable computers and expect to receive several types of feedback, alerts, and guidance. This requires the analysis of objects in the field of vision and the understanding of activities performed by the agents (subjects) in order to predict their intentions.

In the context of this active research field, a relevant interest is to verify whether such technologies can be applied to animal-carried devices. Animal activity monitoring is not an easy task with no humans around. Nevertheless, it is desired for animal behavior researchers to be able to monitor wildlife without strenuously watching hours and hours of video footage. If such data could be obtained and processed in large scale, then it would be possible to make valuable inferences about which animal groups are better adapted to environmental changes, hunting and sleeping routines, among other activities.

Human and animal motion patterns are distinct [4] taking into account moving behavior, biped × quadruped characteristics, and activity motion. A large amount of human actions is done with the hands, whereas animals use their muzzles.

In this paper, we evaluate the descriptive power of Visual Rhythms (VR) [5] in the first-person activity recognition context. It is a method for encoding a video segment into a single image – described in Section 3.1. We use texture description on intensity, gradient, and optical flow VR images to fuse appearance, shape and motion information – and call it VRTD (Visual Rhythm Texture Descriptor). This method is put together using the improved Dense Trajectories [6] framework, introducing a novel scheme to apply these existing methods for a new purpose. We evaluate the proposed method on the DogCentric Activity Dataset [4], a recent first-person dog dataset with realistic videos, and JPL First-Person Interaction Dataset [7], a first person interaction dataset.

The remainder of the text is organized as follows. Section 2 briefly describes the current state of action recognition literature and applications of Visual Rhythms. Section 3 presents and discusses the first-person action recognition methodology proposed in this work. Section 4 reports and evaluates implementation details, the validation dataset, and experimental results. Section 5 concludes the work with final remarks and directions for future work.

2. BACKGROUND

The current state-of-the-art action recognition methods follow the improved Dense Trajectories (iDT) [6, 8] pipeline. The approach consists in a modification of the cuboid construction for a bag-of-word based video classification. Instead of selecting interest points and obtaining their surroundings as a parallelepiped, dense sampled points are tracked over a few frames with optical flow, such that the spatial neighborhood on each position is appended to create a curving volume. These volumes are described with Histograms of Oriented Gradients (HOG), Histograms of Optical Flow (HOF) and Motion Boundary Histograms (MBH), then encoded by Fisher vectors.

Some other works have achieved better accuracies, however, using variations of iDT. In [9], the iDT flow is performed in parallel with the extraction of deep learned two-stream convolutional feature maps [10]. Instead of computing the histogram descriptors, as the original work, each trajectory point was associated with a deep learned feature map location.

Visual Rhythm (VR) [5] is an encoding technique that aims to analyze temporal properties on videos. It consists in transforming each video frame into a single column of a resulting image, that is, a temporal slice. Each pixel of the column corresponds to a spatial structure, and each line in the VR image represents its transformation over time.

Different video analysis tasks have employed visual rhythms. Caption detection is performed in [11] by obtaining the max pooling

The authors are thankful to FAPESP (grants #2015/03156-7 and #2015/12228-1) and CNPq (grants #305169/2015-7 and #307010/2014-7) for their financial support.

over the frame middle columns as the temporal slice. This yields notable rectangles on the produced VR image, which can be easily detected and indicate the caption position and time span. In [12], shot boundaries are identified by detecting sudden changes over several VR lines for video summarization purpose. Face spoofing detection is performed in [13] by constructing the visual rhythm over the Fourier spectrum of residual Moiring effect noises. Gray-level co-occurrence matrices (GLCM) [14] are used to distinguish valid videos from attack videos. In [15], patterns are detected from time series encoded through visual rhythms and used for phenology studies. More details on visual rhythms are given in Section 3.1.

Iwashita et al.[4] approach first person recognition fusing multiple descriptors, local and global. The global descriptors are constructed as grids of optical flow and local binary patterns. The local descriptors used are histograms of optical flow and oriented gradients, together with cuboids [16]. Each local descriptor is coded and pooled separately using bag of visual words and the vectors are fused using multi-channel kernels [17].

A human activity prediction based on dynamic bag-of-words is present by Ryoo [18]. An activity is represented as an integral histogram of spatio-temporal features. The recognition of interaction-level human activities from a first-person view-point is discussed by Ryoo and Matthies [7], where multi-channel kernels are used to integrate local and global motion information.

3. PROPOSED METHODOLOGY

This section presents the main stages of the proposed methodology for action recognition.

3.1. Visual Rhythms

The visual rhythm (VR) image is built by joining slices of all frames of a video. A frame slice is a 1D column image of a set of pixels linearly arranged. There is no constraint in how to choose and arrange the pixels to form a slice. It is arbitrary and depends on what sort of information is desired. Some usage examples are presented in Section 2.

All the slices are appended horizontally to form a $W \times H$ image, where W is the duration, in frames, of the video and H is the size, in pixels, of the slice. This way, each column of a VR image represents an instant in time, whereas each line represents an image pixel or visual structure varying over time.

In this work, we want the entire image to be considered, so that no part of the action is lost. We experimented iterating over the image following a zigzag course. Figure 1 illustrates the construction process of a visual rhythm image.

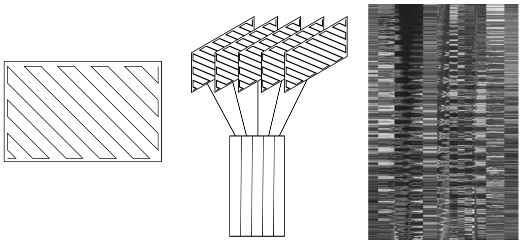
(a) zigzag (b) appending slices (c) example

Fig. 1. Construction of a visual rhythm image. (a) creation of a slice following a zigzag traversal over the frame; (b) concatenation of a sequence of slices to produce a VR image; (c) example of VR image.

An important issue in defining the slice is what type of information is needed. Since our aim is action recognition, we consider four domains:

(i) *original gray scale images:* the original domain carries appearance information.

(ii) *intensity gradients (x and y):* gradients are often used to represent shape and have shown to provide discriminative information for action recognition.

(iii) *optical flow (x and y):* movement information has been shown to be complementary to shape, also contributing with discriminative power.

(iv) *motion boundaries (x and y):* defined as the gradient of optical flow images, they carry information about both shape and movement.

For each video segment, this results in seven visual rhythm images. At a close inspection, it is noticeable that these images resemble texture patterns, as can be seen in Figure 1(c). Therefore, next we ascertain how discriminative texture descriptors are over visual rhythms for action recognition. The proposed VRTD (Visual Rhythm Texture Descriptor) is the concatenation of texture features of every VR image of a given video segment.

3.2. Local Patches

Next, we apply the strategy of Section 3.1 as a descriptor of local patches. Several patches are acquired from a single video. Each patch has its corresponding visual rhythm image built and described (Section 3.3). Figure 2 illustrates this step. Later, they are pooled together into a vector that represents the entire sequence.

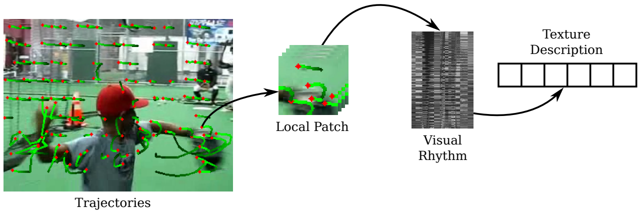

Fig. 2. Construction of the visual rhythm descriptors.

Local patches are obtained through the improved Dense Trajectory (iDT) method [6]. The process consists in densely sampling a set of points of each frame. Every point movement is tracked in the following frames using optical flow. The sequence of space and time coordinates of a point is called a trajectory. The trajectory temporal extension, n_τ, is parameterized and set as 15 frames.

To filter camera motion, each frame is warped in relation to its adjacent previous by a combination of optical flow and speeded up robust feature (SURF) matching. If available, the person's bounding box is excluded from the matching.

To build a local patch of the trajectory, the spatial $N \times N$ surroundings of all trajectory points are concatenated in a volume. This way, instead of extracting parallelepipeds from the video, we build one from a curved volume. This is shown in Figure 3.

3.3. Texture Description

The local volumes are then encoded as visual rhythms and described by their textures to obtain the local descriptors. This texture description over visual rhythms is what we call VRTD, whose representation is the main contribution of this work.

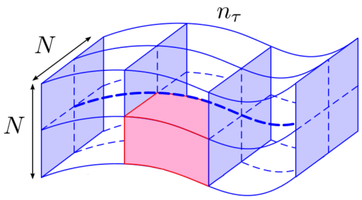

Fig. 3. Local volumes are obtained concatenating all trajectory points spatial surroundings. Image extracted from [8].

Several texture descriptors were considered in our methodology. However, Local Binary Patterns (LBP) [19] were chosen since they achieved promising results and are fast to compute. The LBP method consists in comparing each pixel with their neighbors along a circle. A binary sequence is constructed iterating over the neighboring points and setting the i-th less significant digit to 0 if the central pixel is higher than the i-th neighbor, and 1 otherwise. Then, all pixel descriptors are pooled on a histogram to produce a low-dimensional descriptor for the entire image.

We employ the uniform variation of LBP. A local pattern is considered uniform if it contains no more than two transitions (one-zero or zero-one). For example, the pattern 11111101 has two transitions, so it is uniform, and the pattern 00001010 has four transitions and is not uniform. Every possible uniform pattern has a corresponding bin in the histogram. All non-uniform patterns are assigned to a single bin. An LBP histogram with 8 neighbors has its dimensionality reduced from 256 to 59 with this method.

3.4. Descriptor Pooling

The procedures in Sections 3.1, 3.2 and 3.3 describe feature extraction for only one video patch. To obtain a global descriptor for a video, a pooling strategy is necessary. For this task, we chose Fisher vectors [20], since they have demonstrated to achieve superior results as a global descriptor than the bag-of-word technique for video classification [21].

As described by Perronnin et al. [21], we carried out three steps to enhance the Fisher kernel: L2 normalization, power normalization, and spatial pyramids. As this coding renders further classification kernels unnecessary, we only need to evaluate classifiers with linear kernels. This model is applied separately for each visual rhythm domain and the Fisher vectors are concatenated to form the final descriptors.

4. EXPERIMENTS AND RESULTS

This section presents and discusses the evaluated datasets, experimental setup, and results obtained with the proposed methodology.

4.1. Datasets

Experiments were conducted on two datasets. The first one is the DogCentric Activity Dataset [4], a first-person action dataset centered on dogs. GoPro cameras were installed on the back of four dogs. Their actions were recorded and split into ten categories, totaling 209 videos: playing with a ball, waiting for a car to pass by, drinking water, feeding, turning head to the left, turning head to the right, petting, shaking body, sniffing, and walking. Figure 4 exemplifies these classes.

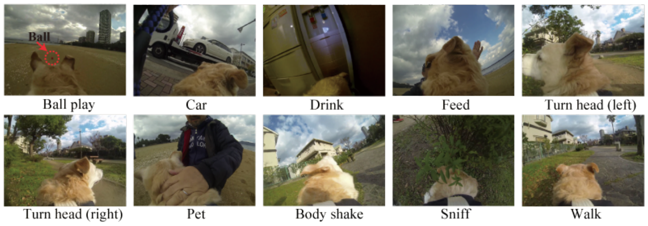

Fig. 4. Ten classes of actions of DogCentric Activity dataset. Extracted from [4].

The second one is the JPL First-Person Interaction Dataset [7], which focuses on group interactions. It contains four positive, or friendly, actions (hand shake, hug, pet and wave), one neutral action (being pointed at or talked about) and two negative, or hostile, actions (punch and throw). Each of these 7 interaction classes are recorded by 12 actors, with a total of 84 videos. Figure 5 shows some examples.

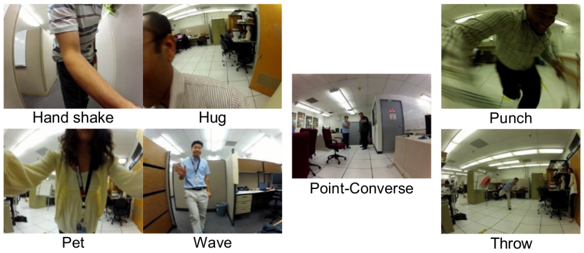

Fig. 5. Seven classes of actions of JPL First-Person Interaction dataset divided into friendly, neutral and hostile. Extracted from [7].

The provided cross validation protocol for both sets are the same: randomly selecting half of the set as training and the other half as testing, then taking the average of multiple runs. We ran the split 30 times.

4.2. Experimental Setup

To evaluate the effectiveness of the VRTD descriptor, we build two setups. The first one tests it globally. The entire videos are used to compose one big visual rhythm image. To reduce data size and processing time, the video frames were rescaled by a factor of 1.7 using bicubic interpolation. The second test follows the improved dense trajectory framework, described in Section 3.2, in which several local patches are obtained from the video. They are independently described with VRTD and pooled together with Fisher vector to compose the final video descriptor.

In both cases, seven visual rhythm images are computed from the videos or patches: gray scale, gradient x and y, optical flow x and y, and motion boundaries x and y. Each image has its texture descriptor extracted using a concatenation of one uniform LBP with radius 1 and 8 neighbors and another with radius 2 and 16 neighbors.

In order to compute the Fisher vectors, we need a probabilistic model; for that, we employ Gaussian Mixture Models (GMM) with 64 mixtures. Before computing the GMM, we transform the local descriptors through Principal Component Analysis (PCA) – we keep just enough dimensions to maintain 99.7% of the variance. Gradients are computed through the Sobel filter and optical flow, according to Farnebäck's method [22].

We use Support Vector Machine (SVM) with linear kernels on the cross-validation experiments, (LIBLINEAR [23] implementation) following the predefined dataset protocol. The resulting fisher vectors are equivalent to using fisher kernel, therefore the linear kernel suffices [21].

The descriptor extraction was implemented in C/C++, using the OpenCV [24] 2.4.12 library. Uniform LBP code was built by fusing the OpenCV source to obtain the binary pattern and Scikit-Image [25] 0.12.3 source to build the uniform patterns, converted to C/C++. Fisher vectors are computed using Yael [26] version 438. Machine learning and data manipulation code was implemented in Python with NumPy [27] 1.10.4, SciPy [28] 0.17.0 and Scikit-Learn [29] 0.17.1 libraries.

4.3. Results

Experiments using the described trajectory-based set-up on the DogCentric Activity dataset achieved accuracy of 69.60%. More than 9% over the baseline [4], which reported an accuracy of 60.5%.

Figure 6 shows the confusion matrix using trajectory VRTD. “Look left” and “drink” classes had the lowest individual accuracy. It matches the baseline, which also had these two actions as its worst: 42.2% and 24%, respectively. This is understandable, since “drink” has similarities to “feed” and “sniff”.

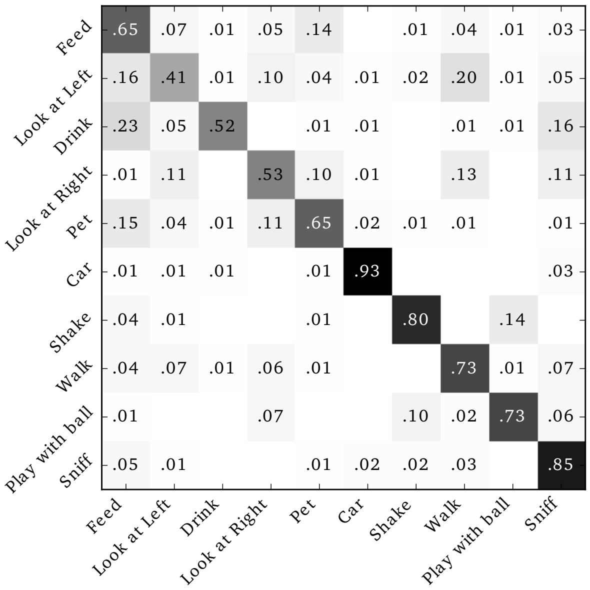

Fig. 6. Confusion matrix for the proposed Trajectory VRTD method on DogCentric Activity dataset [4].

Unexpectedly, the global VRTD approach surpassed the accuracy of the more sophisticated trajectory-based approach. It achieved 84.0% accuracy, which is a little over the baseline [7]. When embedded in the trajectory framework, the accuracy drops considerably to 74.5%.

Figure 7 shows the confusion matrix using global VRTD. Most of the confusion is located between “hug” and “pet” classes, which are both friendly and have relatively high inter-class similarities.

Table 1 summarizes a comparison of our results with the literature on both datasets. The lines marked in bold represent the highest accuracies and both belong to the methods proposed in this work.

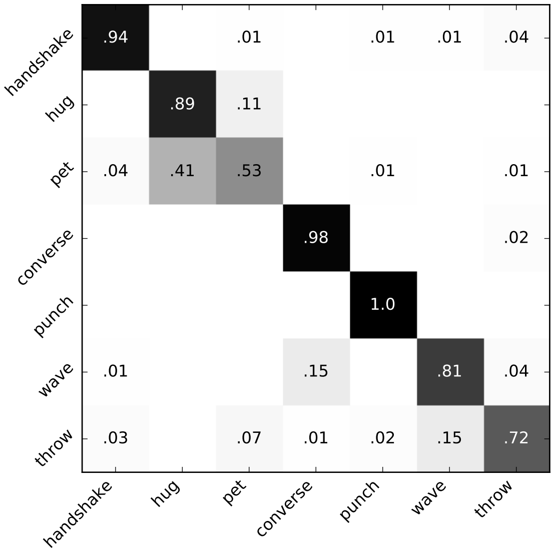

Fig. 7. Confusion matrix for the proposed Global VRTD method on JPL First-Person Interaction dataset [7].

Table 1. Result comparison for DogCentric Activity dataset [4] and JPL First-Person Interaction dataset [7].

	Method	Accuracy (%)
DogCentric	Iwashita et al. [4]	60.5
	Global VRTD	64.5
	Trajectory VRTD	**69.6**
JPL	ST-Pyramid match [30]	82.6
	Dynamic BoW [18]	82.8
	Structure Match [7]	83.1
	Trajectory VRTD	74.5
	Global VRTD	**84.0**

5. CONCLUSIONS AND FUTURE WORK

In this work, we presented the Visual Rhythm Texture Descriptor (VRTD) for first person action recognition. It is obtained as texture features over visual rhythms, as described in Section 3.

We used multiple image domains – grayscale, gradient, optical flow and motion boundaries – and applied them in two manners. One is constructing the visual rhythms using the entire videos and directly describing them using LBP. This approach yielded 84.0% accuracy on the JPL First-Person Interaction dataset [7], which is a little over the baseline. The other one follows the improved dense trajectory approach, building and describing visual rhythms on local patches. This strategy achieved an accuracy of 69.6% on the DogCentric Activity dataset [4], which is superior than the baseline with 60.5%.

Directions for future work include the fusion of VRTD with other descriptors to explore complementary features. We also intend to evaluate the method on other datasets and even other domains. Other texture analysis techniques may yield even better results.

Different local classifier fusion techniques, such as pooled time series (PoT) [17], may also enhance the results by giving the descriptor temporality understanding.

6. REFERENCES

[1] A. Betancourt, P. Morerio, C. S. Regazzoni, and M. Rauterberg, “The Evolution of First Person Vision Methods: A Survey,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 25, no. 5, pp. 744–760, 2015.

[2] S. Narayan, M. S. Kankanhalli, and K. R. Ramakrishnan, “Action and Interaction Recognition in First-person Videos,” in *IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 512–518.

[3] H. Pirsiavash and D. Ramanan, “Detecting Activities of Daily Living in First-person Camera Views,” in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2012, pp. 2847–2854.

[4] Y. Iwashita, A. Takamine, R. Kurazume, and M. S. Ryoo, “First-Person Animal Activity Recognition from Egocentric Videos,” in *22nd IEEE International Conference on Pattern Recognition*, Aug. 2014, pp. 4310–4315.

[5] C. Ngo, T. Pong, and R. Chin, “Detection of Gradual Transitions through Temporal Slice Analysis,” in *IEEE Computer Vision and Pattern Recognition*, vol. 1, 1999, p. 41.

[6] H. Wang and C. Schmid, “Action Recognition with Improved Trajectories,” in *International Conference on Computer Vision*, Dec. 2013, pp. 3551–3558.

[7] M. S. Ryoo and L. Matthies, “First-Person Activity Recognition: What Are They Doing to Me?” in *IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, Jun. 2013.

[8] H. Wang, A. Klser, C. Schmid, and C.-L. Liu, “Dense Trajectories and Motion Boundary Descriptors for Action Recognition,” *International Journal of Computer Vision*, vol. 103, no. 1, pp. 60–79, 2013.

[9] L. Wang, Y. Qiao, and X. Tang, “Action Recognition with Trajectory-Pooled Deep-Convolutional Descriptors,” in *IEEE Computer Vision and Pattern Recognition*, Jun. 2015, pp. 4305–4314.

[10] K. Simonyan and A. Zisserman, “Two-Stream Convolutional Networks for Action Recognition in Videos,” in *Advances in Neural Information Processing Systems 27*, Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, Eds. Curran Associates, Inc., 2014, pp. 568–576.

[11] F. B. Valio, H. Pedrini, and N. J. Leite, “Fast Rotation-Invariant Video Caption Detection Based on Visual Rhythm,” in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer Berlin Heidelberg, 2011, vol. 7042, pp. 157–164.

[12] M. V. M. Cirne and H. Pedrini, “A Video Summarization Method Based on Spectral Clustering,” in *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer Berlin Heidelberg, 2013, vol. 8259, pp. 479–486.

[13] A. S. Pinto, H. Pedrini, W. Schwartz, and A. Rocha, “Video-Based Face Spoofing Detection through Visual Rhythm Analysis,” in *25th SIBGRAPI Conference on Graphics, Patterns and Images)*, Aug. 2012, pp. 221–228.

[14] R. Haralick, K. Shanmugam, and I. Dinstein, “Textural Features for Image Classification,” *IEEE Transactions on Systems, Man and Cybernetics*, vol. 3, no. 6, pp. 610–621, Nov. 1973.

[15] J. Almeida, J. A. dos Santos, B. Alberton, L. P. C. Morellato, and R. S. Torres, “Visual Rhythm-based Time Series Analysis for Phenology Studies,” in *IEEE International Conference on Image Processing*, 2013, pp. 4412–4416.

[16] P. Dollr, V. Rabaud, G. Cottrell, and S. Belongie, “Behavior Recognition via Sparse Spatio-Temporal Features,” in *International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance*, 2005, pp. 65–72.

[17] M. S. Ryoo, B. Rothrock, and L. Matthies, “Pooled Motion Features for First-Person Videos,” in *IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2015.

[18] M. S. Ryoo, “Human Activity Prediction: Early Recognition of Ongoing Activities from Streaming Videos,” in *IEEE International Conference on Computer Vision*, Nov. 2011, pp. 1036i–1043.

[19] T. Ojala, M. Pietikainen, and D. Harwood, “Performance Evaluation of Texture Measures with Classification based on Kullback Discrimination of Distributions,” in *IEEE International Conference on Pattern Recognition*, vol. 1, Oct. 1994, pp. 582–585.

[20] F. Perronnin and C. Dance, “Fisher Kernels on Visual Vocabularies for Image Categorization,” in *Computer Vision and Pattern Recognition*, Jun. 2007, pp. 1–8.

[21] F. Perronnin, J. Sánchez, and T. Mensink, “Improving the Fisher Kernel for Large-Scale Image Classification,” in *European Conference on Computer Vision*. Springer Berlin Heidelberg, 2010, pp. 143–156.

[22] G. Farnebäck, “Two-Frame Motion Estimation Based on Polynomial Expansion,” in *Image Analysis*, ser. Lecture Notes in Computer Science, J. Bigun and T. Gustavsson, Eds. Springer Berlin Heidelberg, 2003, vol. 2749, pp. 363–370.

[23] R.-E. Fan, K.-W. Chang, C.-J. Hsieh, X.-R. Wang, and C.-J. Lin, “LIBLINEAR: A Library for Large Linear Classification,” *Journal of Machine Learning Research*, vol. 9, pp. 1871–1874, 2008.

[24] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.

[25] S. van der Walt, J. L. Schönberger, J. Nunez-Iglesias, F. Boulogne, J. D. Warner, N. Yager, E. Gouillart, T. Yu, and the scikit-image contributors, “Scikit-image: Image Processing in Python,” *PeerJ*, vol. 2, p. e453, 6 2014.

[26] M. Douze and H. Jégou, “The Yael Library,” in *22nd ACM International Conference on Multimedia*. Orlando, FL, USA: ACM, 2014, pp. 687–690.

[27] S. van der Walt, S. C. Colbert, and G. Varoquaux, “The NumPy Array: A Structure for Efficient Numerical Computation,” *Computing in Science & Engineering*, vol. 13, no. 2, pp. 22–30, 2011.

[28] E. Jones, T. Oliphant, and P. Peterson, “SciPy: Open Source Scientific Tools for Python,” 2001. [Online]. Available: http://www.scipy.org/

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, “Scikit-learn: Machine Learning in Python,” *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[30] J. Choi, W. J. Jeon, and S.-C. Lee, “Spatio-temporal Pyramid Matching for Sports Videos,” in *Multimedia Information Retrieval*. Vancouver, BC, Canada: ACM, 2008, pp. 291–297.