International Journal of Engineering & Technology, 7 (4) (2018) 2153-2160

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET
doi: 10.14419/ijet.v7i4.17274
Research paper

Action recognition based on histogram of oriented gradients and spatio-temporal interest points

P. A. Dhulekar [1] *, S. T. Gandhe [1]

[1] Department of E&TC Engineering, Sandip Institute of Technology and Research Centre, India
**Corresponding author E-mail: pravindhulekar@gmail.com*

Abstract

In modern years large extent of the work has been carried out to recognize human actions perhaps because of its wide range of applications in the field of surveillance, human-machine interaction and video analysis. Several methods were proposed by researchers to resolve action recognition challenges such as variations in viewpoints, occlusion, cluttered backgrounds and camera motion. To address these challenges, we propose a novel method comprise of features extraction using histogram of oriented gradients (HOG), and their classification using k-nearest neighbor (k-NN) and support vector machine (SVM). Six different experimentations were carried out on the basis of hybrid combinations of feature extractors and classifiers. Two gold standard datasets; KTH and Weizmann were used for training and testing purpose. The quantitative parameters such as recognition accuracy, training time and prediction speed were used for evaluation. To validate the applicability of proposed algorithm, its performance has been compared with spatio-temporal interest points (STIP) technique which was proposed as state of art method in the domain.

Keywords: *Action Recognition; Histogram of Oriented Gradients; K-Nearest Neighbor; Support Vector Machine; Spatio-Temporal Interest Points.*

1. Introduction

Human action recognition has become key interest for researchers due its wide set of applications in the field of surveillance, controlling and video analysis. The surveillance involves identification of suspicious activities in Banks, ATMs, parking area, shopping stores and many other places where action recognition may plays vital role to increase security of these places [1]. For epileptic patients video monitoring at home is the cost effective solution to identify the type of epilepsy through hand & neck movements of patients [2]. Controlling majorly involves human-machine interaction where the action based controlling of devices can be observed. Many computer applications such as media player, games are controlled by actions; even the computer peripherals such as mouse and keyboard can also be controlled using actions. Video analysis is the part of action recognition applications where specific incident can be search out from long duration or multiple videos. The bowler's movement in cricket can be observed to identify objectionable moves such as nibbling of ball or an act of hugging someone can be traced out from news footage.

Action recognition is the task of classifying the various actions we are performing in our day to day life and assigning them label which can describe it suitably [3]; for example eating, sitting, lifting something, pulling or pushing the doors and many more. While labeling care should be taken in such a way that given label can be understood to average human. Action recognition is the ability of system to identify actions executed by human on the basis of training given that is knowledge based. Therefore accuracy of such system more depends on type of dataset used during training of classifier.

The major bottleneck issues in action recognition are variations in viewpoint, occlusion, cluttered background and camera motion whereas other two challenges such as anthropometric variations and execution rate were completely resolved in existing work [5]. In this paper we propose action recognition methods based on spatio-temporal interest points (STIP) and histogram of oriented gradients (HOG) which are described in section II and III respectively. The k-nearest neighbor (k-NN) and support vector machine (SVM) were used for classification, these classification techniques are discussed in section IV. The proposed methodologies were tested on KTH and Weizmann datasets whose details are specified in section V whereas last section VI focuses on results obtained.

2. Spatio temporal interest points

Interest points are the points that show considerable local deviation of image intensities in spatial domain, these interest points contain significant information of interest. Several applications are based on detection of interest points such as estimation and tracking based on optical flow, stereo matching, and indexing of image.

The interest points are detected in the spatiotemporal domain and obtained space-time features gives key segments of video. To obtain spatio-temporal interest points we adopted the methodology presented by Ivan Laptev and Tony Lindeberg in [6]. To acquire key segments with different spatio-temporal moments, interest points in spatio-temporal scale-space are computed and chosen the scales that approximate to the dimension of the detected segments in space with durations in time.

The spatio-temporal interest points of image can be found by detecting local positive spatio-temporal maxima in H, where H is defined as:

$$H = \lambda_1 \lambda_2 \lambda_3 - k\,(\lambda_1 + \lambda_2 + \lambda_3)^3 \quad (1)$$

In equation (1) $\lambda_1, \lambda_2, \lambda_3$ are eigen values of μ and μ is second-moment matrix. The large value of k gives points with large deviation of the image intensities in spatial and temporal domain.

These features are adaptive to moving patterns size, frequency and velocity due to which it provides stable representation of video. In [7], Christian Schuldt et al. proposed action recognition based on spatiotemporal interest points with local SVM approach. Inline to extension of this work, we combined spatio-temporal interest points with k-NN classification and compared the obtained results with SVM based approach.

3. HOG based action recognition

In the HOG feature descriptor, features are the distribution (histograms) of directions of gradients (oriented gradients). At edges and corners where the change in the intensity is abrupt, this descriptor gives large magnitude of gradients which ultimately provides significant information about the appearance and movements of the subject [8].

Based on Histogram of oriented gradients (HOG), two experimentations have been carried out. In first, features extracted using HOG are classified using k-nearest neighbor (k-NN) and in second, support vector machine (SVM) are used to classify the same. The flow chart of gradients computation has been shown in figure 1 below.

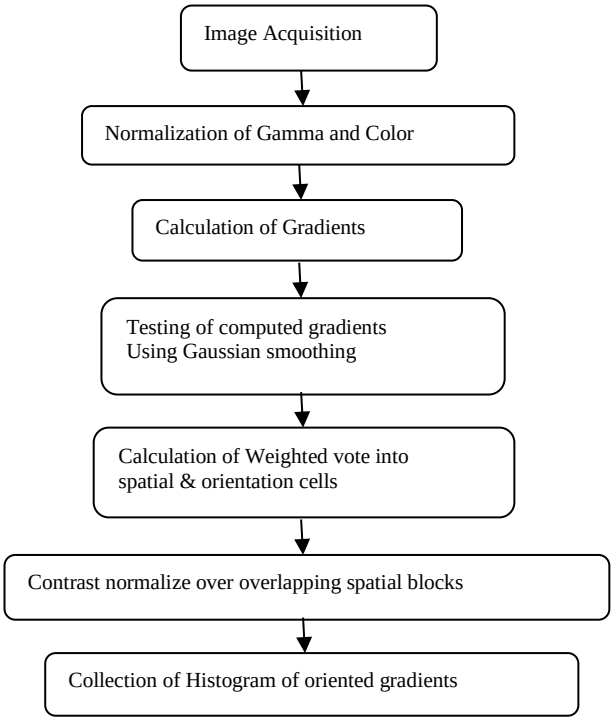
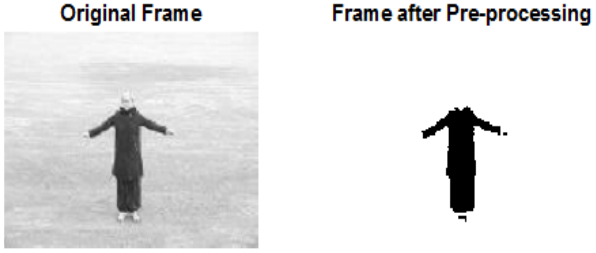


Fig. 1: Process of Gradients Computation.

In this work very first we have loaded training and testing dataset into the directories that are recursively scanned. Thereafter, a preprocessing is done to eliminate noise artifacts raised during collection of the image samples. The preprocessing helps to obtain improved histogram of oriented gradients (HOG) which is observed in figure 2. The gradients are computed from training dataset in third step of the algorithm. Left side of the figure shows HOG of original frame where directions of gradients are very poor whereas right side of the figure 2 shows significant improvement in oriented gradients of preprocessed frame for each cell size.



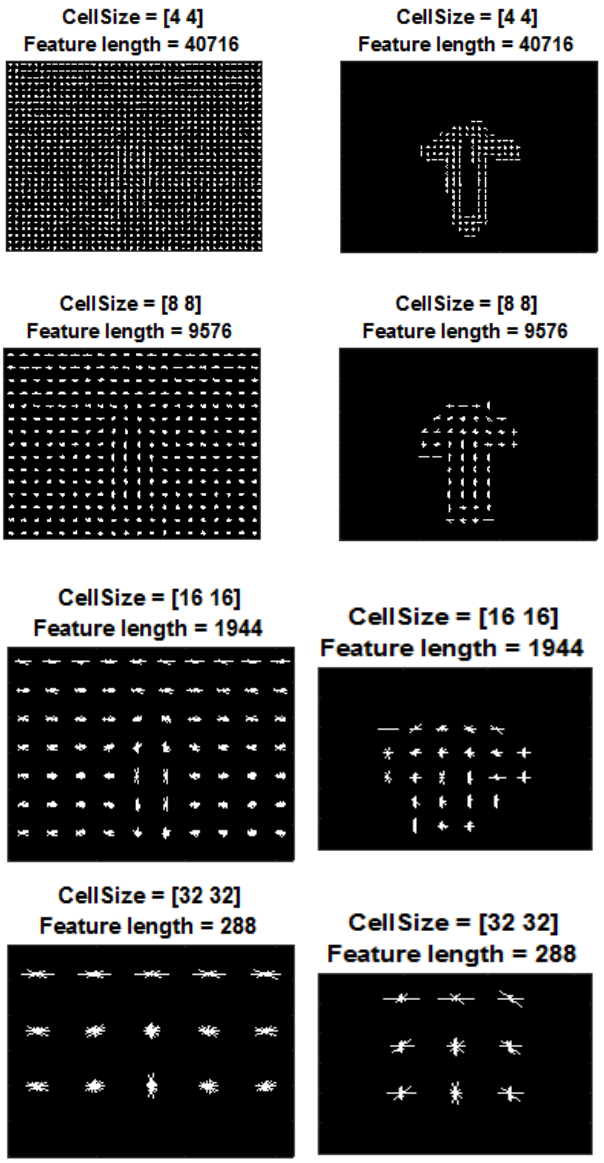


Fig. 2: Original and Preprocessed Frame with Respective Histogram of Gradients Obtained at Different Cell Size.

The change in cell size affects the amount of shape information, accuracy and training speed. The right side of figure 2 shows that a cell size of [32 32] does not encode much shape information, while a cell size of [4 4] encodes a lot of shape information but increases the size of HOG feature vector which slow down the speed of training. Better selection among all cell sizes is 32-by-32 due to its speed and recognition accuracy calculated over 280 key frames of Weizmann dataset. The details are depicted in table I.

Table 1: Effect of Cell Size on Speed of Operation

Cell size	Dimension	Block	Bins	Speed (sec)	Accuracy (%)
4 x 4	40716	2 X 2	9	> 1 Hr	18.69
8 x 8	9576	2 X 2	9	799.96	20.71
16 x 16	1944	2 X 2	9	237.87	23.92
32 x 32	288	2 X 2	9	103.03	32.14

The extracted features are used to train k-NN and SVM classifier, here SVM applied for classification is multiclass. Once the training is finished, in the last step these fitted classifiers are used to make predictions about input video sequences to identify their category (Label).

4. Classification techniques

To classify the features extracted by the method of spatio-temporal interest points and histrogram of oriented gradients as described in II and III sections, supervised learning algorithms such as k-nearest neighbor and support vector machine were used. The supervised learning takes a known set of input data and known responses to the data, and seeks to build a predictor model that generates reasonable predictions for the response to new data.
The classifier selected on the basis of characteristics such as Predictive Accuracy, Fitting Speed, Prediction Speed and Memory Usage. There are always tradeoffs between these characteristics. Comparing k-NN with SVM, predictive accuracy and fitting speed of k-NN is high only for low dimension data while it is poor in case of high dimension whereas SVM gives high accuracy and medium fitting speed in all cases. Prediction speed and memory usage for SVM are good if there are few support vectors, but can be poor if support vectors are many, whereas k-NN has medium prediction speed with high memory usage.

4.1. k-nearest neighbor (k-NN)

Training the nearest neighbor classifier need known input data and known responses, in our case known input data contains 50 observations each with 40 predictors for single class or category of action. This known input data are the features extracted for each action using STIP and HOG. The known responses of data are categorical vectors containing 6 different action labels boxing, handclapping, handwaving, jogging, running and walking of KTH dataset, each with 50 entries corresponding to 50 observations. Similar work is extended for Weizmann dataset.
The predicted responses are obtained through new data and trained model based on new data points distance from points in a training dataset. In k-NN various metrics are used for distance computation such as Euclidian, standardized Euclidian, Mahalanobis, City block, Minkowski, Chebychev, Cosine, Correlation, Hamming, Jaccard and Spearman. For this work, standardized euclidian with three nearest neighbors has been selected which was determined after conducting the several tests with different number of neighbors and distance metrics. The computation formula for standardized euclidian distance is given in equation 2.

$$d^2_{ij} = (x_i - y_j)\, D^{-1}\, (x_i - y_j)' \tag{2}$$

Where x_i represents training dataset points, y_j represents testing dataset points and D is the diagonal matrix whose nth diagonal element is $M(n)^2$, where M is inverse weights vector.
Though classification using k-NN is quite simple but its major drawback is its performance degrades in case of high dimensional data.

4.2. Support vector machine (SVM)

In SVM classification technique, the hyperplane is designed to separates all data points of one class from those of the other class. The hyperplane that provides largest margin between the two classes is most suitable for classification. Margin means the maximal width of the slab parallel to the hyperplane that does not have data points inside it [9].
The support vectors are defined as data points that are closest to the separating hyperplane; these points are located on the boundary of the slab. Figure 3 illustrates these definitions, where type 1 data points are indicated by '+' while type 2 data points are indicated by '-'

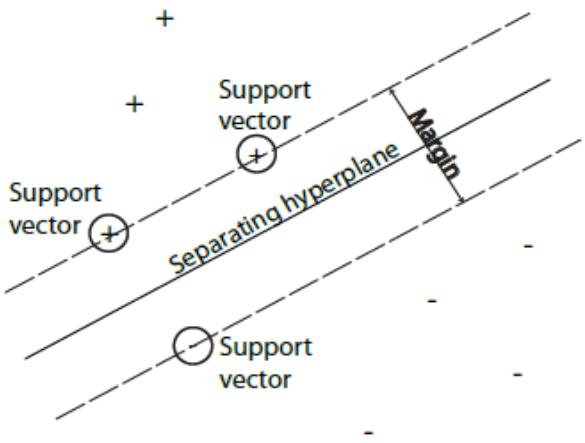


Fig. 3: Concept of Hyperplane.

In case of separable data where all data points can be easily separated by hyperplane, the separating hyperplane is equated as:

$$F(x) = x'\beta + b = 0 \tag{3}$$

Where $\beta \in R^d$, for dimension d the set of points $x \in R^d$ and b is a real number.
In case of non separable data, hyperplane cannot separate all data points therefore soft margin is adopted for separating hyperplane which is defined in equation (4).

$$\min_{\beta,b,\xi}\left(\frac{1}{2}\beta'\beta + C\sum_j \xi_j\right) \tag{4}$$

Such that

$$y_j f(x_j) \geq 1 - \xi_j$$
$$\xi_j \geq 0.$$

Where ξj is slack variables and C is a penalty parameter.
In this work, an error-correcting output codes (ECOC) multiclass model is trained using multiple binary support vector machine (SVM). Feature matrix and labels corresponds to each observation in feature matrix were used to train ECOC model. The predictors of feature matrix were standardized before training and classifier is cross validated to reduce the classification error.

5. Dataset

Use of proper dataset for training and testing is the key aspect for every action recognition algorithm, as the results of every algorithm depends on conditions under which dataset has been recorded. Various datasets available for human action recognition are KTH, Weizmann, ICS Action Database, Korea University Gesture Database, INRIA XMAS, Wearable Action Recognition Database, CASIA motion, Biological Motion Library, HDM05 Motion Capture Database, Cambridge Gesture Dataset, NATOPS Dataset, Keck gesture datasets, UCF sports, HOHA, YouTube dataset, VIRAT Video Dataset, MSR Action Dataset, UTexas Database, Human Eva-I/ -II Database, CMU Mocap Database, Human Motion Database (HMD), Interactive Emotional Dyadic MoCo DB (IEMOCAP) Database, MuHAVi, , CHIL 2007 Evaluation dataset, DLSBP dataset, Manually Annotated Silhouette Data (MAS), Virtual Human Action Silhouette (ViHASi) Data, POETICON Enacted Scenario Corpus, TMU Kitchen Dataset, Kitchen Capture Dataset, , West Virginia university multi-view (WVU), ChaLearn Gesture, G3D a gaming Dataset, Assisted Daily Living (ADL) and i3DPost Multi-view dataset [4]
Here preferably we have selected KTH and Weizmann dataset due to the wide range of variation introduced through camera motion,

anthropometric variations, different clothing, and dynamic background. These datasets has addressed most of the action recognition challenges. The detail characteristics of these dataset are discussed in following subsections.

5.1. KTH dataset

The KTH human action dataset has six types of human actions including walking, running, jogging, boxing, hand-waving, and hand-clapping; each type of action is performed by 25 actors in four different conditions: indoors, outdoors, outdoors with variation in scale and outdoors with dissimilar clothing. Dataset contains 600 video sequences. All video sequences were taken over homogeneous backgrounds with a static camera with 25fps frame rate. The sequences were downsampled to the spatial resolution of 160x120 pixels and have a length of four seconds in average [10]. Camera motion has been introduced in the dataset by zooming of the camera. Single template of each action under KTH dataset has been shown in figure 4.

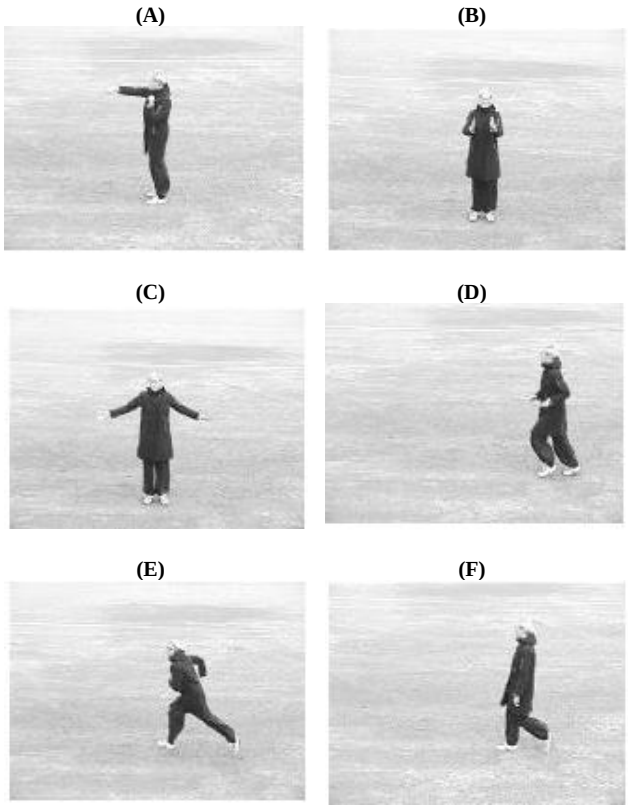


Fig. 4: KTH Frames of (A) Boxing (B) Handclapping (C) Handwaving (D) Jogging (E) Running (F) Walking.

5.2. Weizmann dataset

It consists of 90 low-resolution (180x144, de-interlaced 50 fps) videos of 9 different subjects, each performing 10 natural actions: bend, jumping jack, jump forward, jump in place, run, gallop sideways, skip, walk, wave one hand and wave both hands [11]. This dataset uses a fixed camera setting and a simple background. Holistic features are more suitable for Weizmann. Figure 5 shows the sample frames of each action under Weizmann dataset.

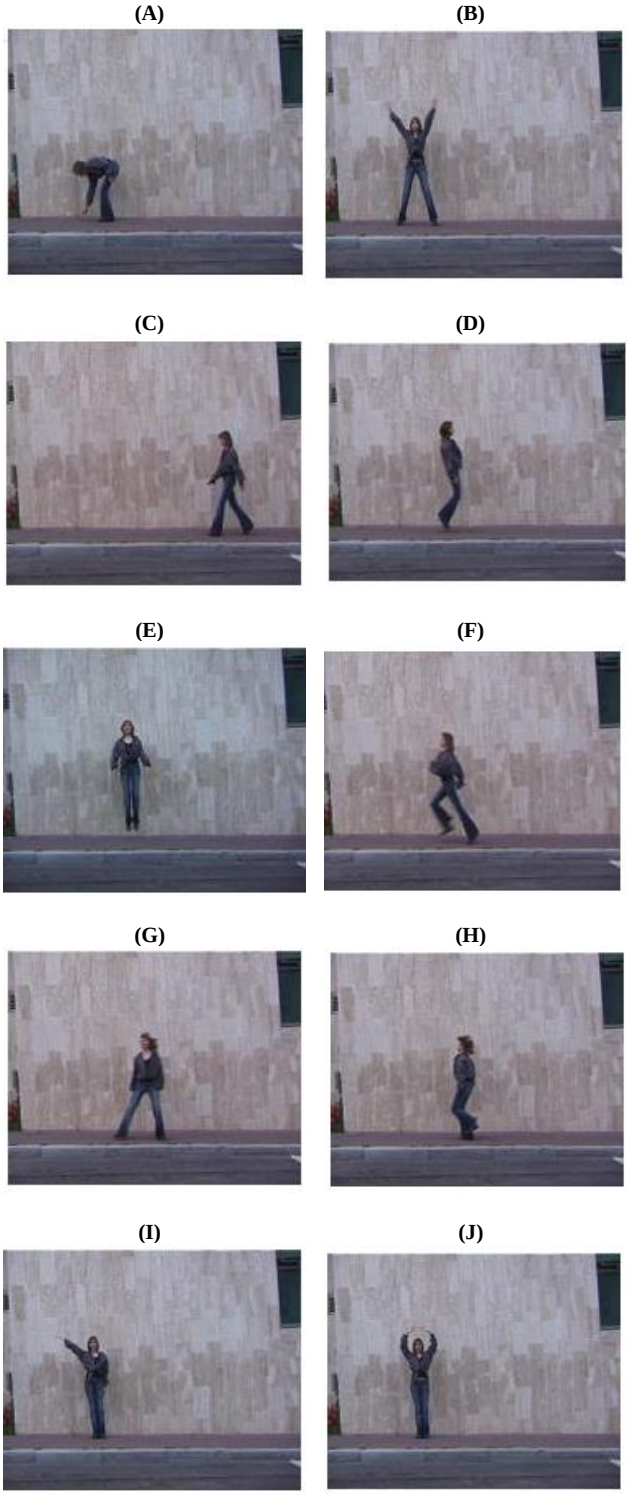


Fig. 5: Weizmann Frames of (A) Bend (B) Jack (C) Walk (D) Jump (E) Jump in Place (F) Run (G) Gallop Sideways (H) Skip (I) Wave One Hand (J) Wave Both Hands.

6. Results and observations

Three types of experimental evaluations have been carried, one with Weizmann datasets and two with KTH Datasets. Four proposed algorithms having the combination of two feature extraction and two classification methods were tested with the help of these evaluations. The performance of algorithms has been measured on the basis of four major parameters; Recognition Accuracy, Feature extraction time, Training time and Prediction speed which are also referred as testing time.

The recognition accuracy is the ratio of number of templates whose predicted label is matched with test label to total number of templates. The formula can be specified as:

$$\text{Accuracy (\%)} = \frac{\text{No. of True Label Templates}}{\text{Total No. of Templates}} \times 100 \quad (6)$$

Table 2: Recognition Accuracy Obtained on Weizmann

Type of Case	Training Data	Testing Data	STIP with k-NN			STIP with SVM			HOG with k-NN			HOG with SVM		
			True (/280)	False (/280)	Accuracy (%)	True (/280)	False (/280)	Accuracy (%)	True (/280)	False (/280)	Accuracy (%)	True (/280)	False (/280)	Accuracy (%)
Case I	S2,S3,S4,S5,S6,S7,S8,S9	S1	76	204	27.14	90	190	32.14	68	212	32.07	90	190	32.14
Case II	S1,S3,S4,S5,S6,S7,S8,S9	S2	54	226	19.28	59	221	21.07	142	138	50.71	157	123	56.07
Case III	S1,S2,S4,S5,S6,S7,S8,S9	S3	73	207	26.07	96	184	34.28	189	91	67.50	172	108	61.42
Case IV	S1,S2,S3,S5,S6,S7,S8,S9	S4	56	224	20.00	98	182	35.00	116	164	41.42	146	134	52.14
Case V	S1,S2,S3,S4,S6,S7,S8,S9	S5	52	228	18.57	58	222	20.71	89	191	31.78	71	209	25.35
Case VI	S1,S2,S3,S4,S5,S7,S8,S9	S6	76	204	27.14	79	201	28.21	177	103	63.21	167	113	59.64
Case VII	S1,S2,S3,S4,S5,S6,S8,S9	S7	55	225	19.64	75	205	26.78	124	156	44.28	141	139	50.35
Case VIII	S1,S2,S3,S4,S5,S6,S7,S9	S8	57	223	20.35	51	229	18.21	178	102	63.57	187	93	66.78
Case IX	S1,S2,S3,S4,S5,S6,S7,S8	S9	56	224	20.00	71	209	25.35	179	101	63.92	156	124	55.71
AVERAGE			62	218	22.14	76	204	27.14	141	139	50.94	143	137	51.06

Table 3: Response Time Obtained on Weizmann

Type of Case	Training Data	Testing Data	STIP with k-NN Time (Seconds)			STIP with SVM Time (Seconds)			HOG with k-NN Time (Seconds)			HOG with SVM Time (Seconds)		
			Feature Extraction	Training Time	Pred. Speed	Feature Extraction	Training Time	Pred. Speed	Feature Extraction	Training Time	Pred. Speed	Feature Extraction	Training Time	Pred. Speed
Case I	S2,S3,S4,S5,S6,S7,S8,S9	S1	72.38	0.32	0.21	72.34	6.40	0.11	21.54	0.23	0.42	23.72	3.90	0.18
Case II	S1,S3,S4,S5,S6,S7,S8,S9	S2	74.31	0.05	0.04	73.63	6.59	0.11	21.52	0.19	0.38	28.19	4.36	0.19
Case III	S1,S2,S4,S5,S6,S7,S8,S9	S3	72.17	0.62	0.20	94.49	10.14	0.22	20.95	0.14	0.42	24.33	4.37	0.19
Case IV	S1,S2,S3,S5,S6,S7,S8,S9	S4	74.12	0.04	0.05	72.07	5.97	0.10	22.98	0.15	0.44	23.66	3.95	0.19
Case V	S1,S2,S3,S4,S6,S7,S8,S9	S5	75.74	0.04	0.04	80.67	6.82	0.11	22.33	0.12	0.39	24.75	4.00	0.18
Case VI	S1,S2,S3,S4,S5,S7,S8,S9	S6	76.15	0.04	0.04	79.82	6.74	0.11	21.72	0.12	0.39	24.96	4.38	0.18
Case VII	S1,S2,S3,S4,S5,S6,S8,S9	S7	73.70	0.33	0.21	76.97	8.47	0.25	20.86	0.15	0.42	25.48	3.74	0.18
Case VIII	S1,S2,S3,S4,S5,S6,S7,S9	S8	72.14	0.04	0.04	77.88	5.46	0.11	21.71	0.13	0.39	25.87	4.43	0.20
Case IX	S1,S2,S3,S4,S5,S6,S7,S8	S9	74.20	0.89	0.28	72.65	9.73	0.17	48.64	1.74	0.65	24.50	4.40	0.19
AVERAGE			73.87	0.26	0.12	77.83	7.36	0.14	24.69	0.33	0.43	25.05	4.17	0.18

Feature extraction time is the time taken by algorithm to extract the features of all data required for training and testing. Training time is the time required for fitting the model by using two input arguments that are set of input data and known responses to the data (output). Prediction speed is speed at which trained (fitted) model generates the reasonable predictions for the response to new data. It can be also called as testing time.

6.1. Experimental evaluation on Weizmann

In this evaluation proposed four algorithms; STIP with k-NN, STIP with SVM, HOG with k-NN and HOG with SVM were tested on Weizmann datasets. To enhance the training and testing speed a concept of action snippets has been adopted [12].

In this experimentation, temporal Segmentation of 90 Videos (10 actions performed by 9 subjects) has been done, among the segmented videos shortest length video sequence has been identified which contains 28 frames. To provide unbiased evaluation, all sequences were trimmed down to 28 frames same as the length of shortest sequence. Shortening of sequence is achieved through elimination of subject less frames and periodicity in actions.

Leave one- out cross-validation is used for all evaluations in which 8 subjects are used for training that constitutes 2240 frames (28 frames x 10 actions x 8 subjects) whereas one subject is used for testing which constitutes 280 frames (28 frames x 10 actions x 1 subject). This process is repeated for all nine combinations, and obtained results were averaged.

In Table 2 & 3, total 9 cases are shown which represents the 9 permutations as per the subjects for training and testing. The last row of each table represents the average results. From results shown in Table 2 it can be observed that HOG with k-NN & SVM provides the recognition accuracy of 50.94% & 51.06% which is almost twice of STIP with k-NN & SVM i.e. 22.14% & 27.14%. Here Table 3 depicts results of response time required for feature extraction, training and testing (prediction speed) of Weizmann dataset. Time required for extraction of STIP features (73.87s & 77.83s) is three times that of HOG features (24.69s & 25.05s) in case of both the classifiers. Along with SVM classifier both methods STIP & HOG takes large time for training (7.36s & 4.17s) compare to when they are combined with k-NN (0.26s & 0.33s) whereas prediction speeds of all methods are approximately equal that is 0.12s, 0.14s, 0.33s and 0.18s.

6.2. Experimental evaluation on KTH (type I)

In this evaluation four algorithms as detailed in previous subsection were tested on KTH dataset. Two types of experimentation were carried out on KTH dataset on the basis of subject and scenario categories. For KTH Type-I experimentation category is subject whereas for KTH Type-II experimentation category is scenario.

Under this evaluation, temporal Segmentation of all 600 Videos (6 actions by 25 subjects in 4 different scenarios) has been done. Here the shortest length has been identified as 18 frames which are sufficient to represent single cycle of action with visibility of subject in each frame.

Table 4: Recognition Accuracy Obtained on KTH (Type I)

Type of Case	Training Data	Testing Data	STIP with k-NN			STIP with SVM			HOG with k-NN			HOG with SVM		
			True (/2160)	False (/2160)	Accuracy (%)	True (/2160)	False (/2160)	Accuracy (%)	True (/2160)	False (/2160)	Accuracy (%)	True (/2160)	False (/2160)	Accuracy (%)
Case I	S6-S25	S1-S5	404	1756	18.70	429	1731	19.86	1144	1016	52.96	1155	1005	53.47
Case II	S1-S5&S11-S25	S6-S10	362	1798	16.75	404	1756	18.70	1243	917	57.54	1284	876	59.44

Case III	S1-S10&S16-S25	S11-S15	364	1796	16.85	323	1837	14.95	999	1161	46.25	1062	1098	49.16
Case IV	S1-S15&S21-S25	S16-S20	409	1751	18.93	440	1720	20.37	1052	1108	48.70	1204	956	55.74
Case V	S1-S20	S21-S25	338	1822	15.64	413	1747	19.12	1074	1086	49.72	1023	1137	47.36
AVERAGE			376	1784	17.40	402	1758	18.61	1103	1058	51.06	1146	1014	53.05

Table 5: Response Time Obtained on KTH (Type I)

Type of Case	Training Data	Testing Data	STIP with k-NN Time (Seconds)			STIP with SVM Time (Seconds)			HOG with k-NN Time (Seconds)			HOG with SVM Time (Seconds)		
			Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed
Case I	S6-S25	S1-S5	239.24	0.51	1.37	340.07	119.55	0.44	87.79	0.29	8.98	88.06	282.79	0.63
Case II	S1-S5&S11-S25	S6-S10	238.78	0.12	1.01	355.67	94.63	0.32	89.57	0.88	8.19	80.91	257.35	0.43
Case III	S1-S10&S16-S25	S11-S15	261.80	0.12	1.12	368.05	89.80	0.32	84.83	0.29	7.64	85.55	247.67	0.44
Case IV	S1-S15&S21-S25	S16-S20	239.10	0.11	1.05	363.68	87.38	0.31	82.74	0.29	7.50	84.50	261.93	0.56
Case V	S1-S20	S21-S25	236.46	0.13	1.01	328.74	97.16	0.30	84.91	0.29	7.63	85.82	256.68	0.44
AVERAGE			243.07	0.19	1.11	351.24	97.70	0.33	85.96	0.40	7.98	84.96	261.28	0.50

Table 6: Recognition Accuracy Obtained on KTH (Type II)

Type of Case	Training Data	Testing Data	STIP with k-NN			STIP with SVM			HOG with k-NN			HOG with SVM		
			True (/2700)	False (/2700)	Accuracy (%)	True (/2700)	False (/2700)	Accuracy (%)	True (/2700)	False (/2700)	Accuracy (%)	True (/2700)	False (/2700)	Accuracy (%)
Case I	d2,d3,d4	d1	566	2134	20.96	482	2218	17.85	1549	1151	57.37	1659	1041	61.44
Case II	d1,d3,d4	d2	601	2099	22.25	484	2216	17.92	1215	1485	45.00	1257	1443	46.55
Case III	d1,d2,d4	d3	497	2203	18.40	462	2238	17.11	1372	1328	50.81	1480	1220	54.81
Case IV	d1,d2,d3	d4	451	2249	16.70	517	2183	19.14	719	1981	26.62	961	1739	35.59
AVERAGE			529	2171	19.59	487	2213	18.03	1214	1486	44.96	1340	1360	49.62

Table 7: Response Time Obtained on KTH (Type II)

Type of Case	Training Data	Testing Data	STIP with k-NN Time (Seconds)			STIP with SVM Time (Seconds)			HOG with k-NN Time (Seconds)			HOG with SVM Time (Seconds)		
			Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed	Feature Extraction	Training Time	Prediction Speed
Case I	d2,d3,d4	d1	276.93	0.11	1.20	365.22	87.07	0.38	83.92	0.29	9.21	181.98	246.53	0.70
Case II	d1,d3,d4	d2	273.13	0.12	1.18	409.12	86.59	0.39	84.18	0.27	8.82	203.39	187.79	0.54
Case III	d1,d2,d4	d3	275.88	0.11	1.17	276.34	73.52	0.39	90.15	0.28	8.74	113.53	194.33	0.56
Case IV	d1,d2,d3	d4	274.69	0.11	1.22	269.50	81.73	0.42	84.47	0.31	8.87	83.43	200.98	0.55
AVERAGE			275.15	0.11	1.19	330.04	82.22	0.39	85.68	0.28	8.91	145.58	207.40	0.58

Five-fold cross-validation is used for all evaluations in which data is segregated into 5 folds, each containing 5 subjects. 4 folds are used for training that constitutes 8640 frames (18 frames x 6 actions x 20 subjects x 4 scenarios) whereas 1 fold is used for testing which constitutes 2160 frames (18 frames x 6 actions x 5 subjects x 4 scenarios). This process is repeated for all five permutations, and obtained results were averaged.

In Table 4 & 5, total 5 cases are shown as per the group of different 5 subjects each time for testing. The last row of each table represents the average results. From results shown in Table 4 it is observed that HOG with kNN & SVM provides the recognition accuracy of 51.06% & 53.05% which is three times that of STIP with k-NN & SVM i.e. 17.40% & 18.61%. Table 5 depicts results of response time required for feature extraction, training and testing (prediction speed) of KTH Type-I. Time required for extraction of STIP features are 243.07s & 351.24s which is also three times more of HOG features i.e. 85.96s & 84.96s. Along with SVM classifier both methods STIP & HOG takes large time for training (97.70s & 261.28s) compare to when they are combined with k-NN (0.19s & 0.40s) whereas predictions speed of k-NN (1.11s & 7.98s) is less than SVM (0.33s & 0.50s).

6.3. Experimental evaluation on KTH (Type II)

In this experimentation the number of videos, type of segmentation and length of shortest sequence is same as KTH Type-I, but the cross-validation method and base to create fold different.

All evaluations were done with 4-fold cross-validation: the data is split into 4 folds of 4 scenarios (indoors, outdoors, outdoors with variation in scale and outdoors with dissimilar clothing) each, 3 folds are used for training which contains 8100 frames (18 frames x 6 actions x 25 subjects x 3 scenarios), 1 for testing which contains 2700frames(18framesx6actionsx25subjects x1 scenarios).

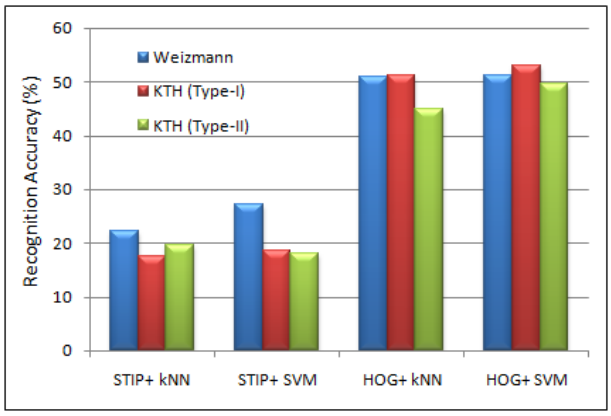


Fig. 6: Plot of Comparative Analysis of Recognition Accuracy (%).

The results are averaged over the 4 permutations. In Table 6 & 7 four cases are represented as per the 3:1 ratio of scenarios for training and testing. The last row of each table represents the average results. From results of Table 6 it is observed that HOG with k-NN & SVM provides recognition accuracy of 44.96% & 49.62% which is more than double of STIP with k-NN & SVM i.e. 19.59% & 18.03%. Table 7 depicts results of response time required for feature extraction, training and testing (prediction speed) of KTH

Type-II. Time required for extraction of STIP features (275.15s & 330.04s) is 2-3 times that of HOG features (85.68s & 145.58s) in case of both the classifiers. Along with SVM classifier both methods STIP & HOG takes significantly large time for training (82.22s & 207.40s) compare to when they are combined with k-NN (0.11s & 0.28s) whereas SVM with predictions speed of 0.39 & 0.58 seconds is notably faster than k-NN with 1.19 & 8.91 seconds.

6.4. Summary

By looking towards the algorithms implemented, results obtained on the basis of different performance measures and types of dataset used for the experimentation, it is necessary to summarize the overall results to notify the exact outcomes of the work.

To provide detail insight into the results of accuracy, we have differentiated recognition accuracy at template level and at video level. The template level accuracy is defined in equation (6) whereas video level accuracy is ratio of number of true labels given to complete video (unlike template) to total number of videos. Table 8 shows the average recognition accuracy obtained at template and video level where it is observed that in all types of datasets HOG performs 2-3 times better than STIP. Even at video level, in case of KTH dataset HOG provides more than 90% recognition accuracy while maximum accuracy of STIP at video level is limited to 31.11% only and at template level it doesn't

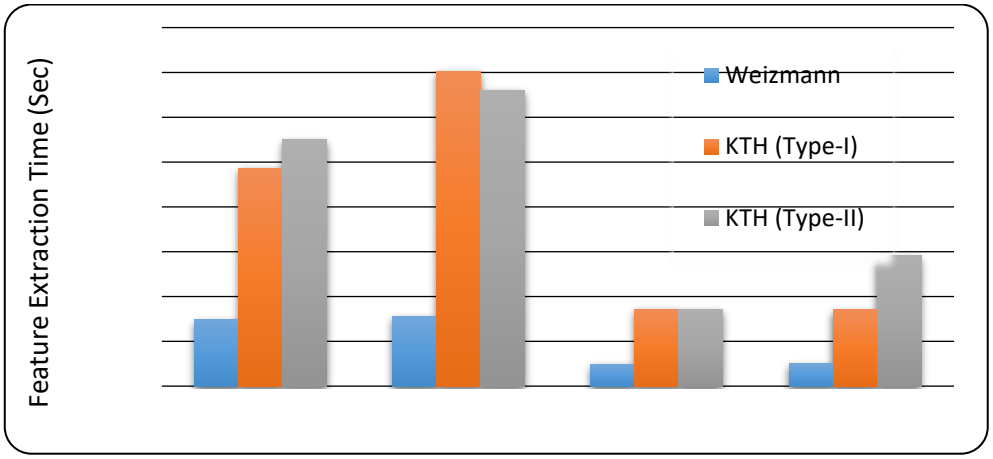


Fig. 7: Plot of Comparative Analysis of Feature Extraction Time (Sec).

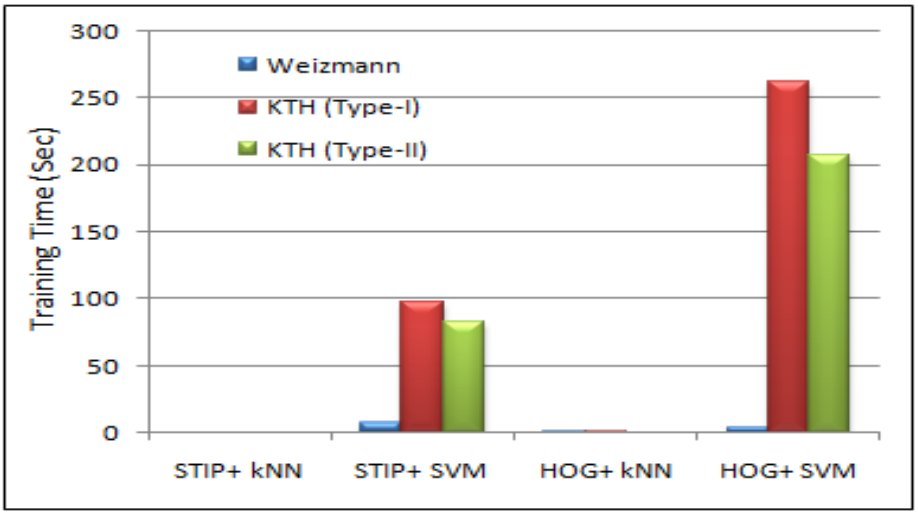


Fig. 8: Plot of Comparative Analysis of Training Time (Seconds).

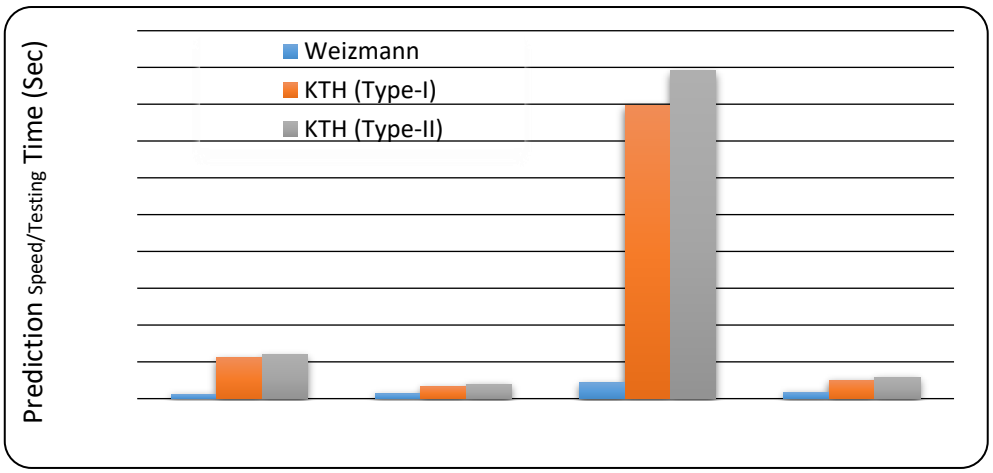


Fig. 9: Plot of Comparative Analysis of Testing Time (Seconds).

Table 8: Average Recognition Accuracy (Template vs. Video)

Method	Average Recognition Accuracy (%) Weizmann		KTH (Type I)		KTH (Type II)	
	Template Level	Video Level	Template Level	Video Level	Template Level	Video Level
STIP with kNN	22.14	27.77	17.40	16.67	19.59	29.17
STIP with SVM	27.14	31.11	18.61	13.33	18.03	16.67
HOG with kNN	50.94	67.77	51.06	100.00	44.96	83.33
HOG with SVM	51.06	57.77	53.05	93.33	49.62	91.66

Table 9: Total Response Time

Method	Total Response Time (sec) [Includes time for feature extraction, training and testing]		
	Weizmann	KTH (Type I)	KTH (Type II)
STIP with kNN	74.26	244.38	276.46
STIP with SVM	85.34	449.28	412.66
HOG with kNN	25.45	94.36	94.87
HOG with SVM	29.40	346.75	353.57

Exceed beyond 30%. The figure 6 shows the plot of comparative analysis of recognition accuracy (%) of all experimental evaluation done at template level.

Comparative analysis also done on the basis of total time taken by algorithm right from data acquisition to action labeled which includes feature extraction, training and testing time. The result of this comparative analysis is shown in Table 9. In this case HOG with k-NN provides better results among all four combinations. It's due to fast training speed of nearest neighbor classifier. Comparison is done separately between feature extraction and classification methods. In case of feature extraction time, HOG is 2-3 times faster than STIP which can be noticed in figure 7. The plots shown in figure 8 and 9 are more relative to performance of classifier than feature extraction where it is observed that k-NN takes least training time whereas SVM takes least testing time.

7. Conclusion

This paper provides comprehensive comparative analysis of two well known feature extraction methods; spatio-temporal interest points (STIP) and histogram of oriented gradients (HOG) on the basis of their performance for human action recognition which is one of the fascinating research areas of recent years. Analysis also compares performance of two classification techniques k-nearest neighbor (k-NN) and support vector machine (SVM) to identify most suitable classifier for action recognition. The comparative analysis has been carried out with some quantitative parameters such as recognition accuracy, feature extraction time, training time and prediction speed that are recorded during testing of proposed algorithms on widely used dataset such as KTH and Weizmann.

Altogether this analysis shows outperformance of histogram of oriented gradients over spatio-temporal interest points in terms of all performance measures. Concluding the performance of classification techniques, it is observed that in most of the cases SVM provided better accuracy over k-NN whereas response time of k-NN is less in all cases compared to SVM. Overall, this work is attempt towards selection of appropriate feature extraction and classification technique for human action recognition. Still there is further scope for development of more robust method that will not only improve recognition accuracy and response time but also resolves all recognition challenges such as variations in viewpoints, occlusion, cluttered backgrounds and camera motion effectively.

References

[1] H. Liu, S. Chen, and N. Kubota, "Intelligent video systems and analytics: ASurvey," IEEE Trans. Ind. Informat., vol. 9, no. 3, pp. 1222–1233, Aug. 2013. https://doi.org/10.1109/TII.2013.2255616.

[2] K. Cuppens, L. Lagae, B. Ceulemans, S. Van Huffel and B. Vanrumste, "Automatic video detection of body movement during sleep based on optical flow in pediatric patients with epilepsy," Medical and Biological Engineering and Computing, Vol. 48(9), pp. 923-931, 2010. https://doi.org/10.1007/s11517-010-0648-4.

[3] Daniel Weinland, Remi Ronfard, Edmond Boyer, "A survey of vision-based methods for action representation, segmentation and recognition," Computer Vision and Image Understanding 115, pp. 224–241, 2011. https://doi.org/10.1016/j.cviu.2010.10.002.

[4] Md. Atiqur Rahman Ahad, J. Tan, H. Kim, S. Ishikawa, "Action Dataset – A Survey," SICE Annual Conference, Waseda University, Tokyo, Japan, September 13-18, 2011.

[5] Manoj Ramanathan, Wei-Yun Yau, and Eam Khwang Teoh, "Human Action Recognition with Video Data: Research and Evaluation Challenges," IEEE Transactions on Human-Machine Systems, Volume: 44, Issue: 5, **pp** 650 – 663, Oct. 2014.

[6] I. Laptev and T. Lindeberg, "Space-time interest points," In Proc. of International Conference of Computer Vision, pp.432–439, 2003. https://doi.org/10.1109/ICCV.2003.1238378.

[7] Christian Schuldt, Ivan Laptev,Barbara Caputo, "Recognizing Human Actions: A Local SVM Approach," In Proc. of the 17th International Conference on Pattern Recognition (ICPR'04), pp.1-5, 2004.

[8] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," In CVPR, pp: 886–893, 2005.

[9] Christianini, N., and J. Shawe-Taylor, "An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods," Cambridge University Press, Cambridge, UK, 2000. https://doi.org/10.1017/CBO9780511801389.

[10] http://www.nada.kth.se/cvap/actions/

[11] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, Ronen Basri, Actions as space–time shapes, in: Proceedings of the International Conference on Computer Vision (ICCV'05), vol. 2, pp. 1395–1402, October 2005.

[12] K. Schindler and L. van Gool, "Action Snippets: how many frames does human action recognition require?" in Proc. IEEE Conf. Comput. Vis.Pattern Recognition, Jun. 2008, pp. 1–8.