

Tri-level Unified Framework for Human Gait Analysis

S. Nissi Paul¹, Yumnam Jayanta Singh²

^{1,2}Dept. Computer Science and Engineering & Information Technology,
School of Technology, Assam Don Bosco University
Guwahati, Assam – India

¹nissi.paul[@]dbuniversity.ac.in

²jayanta[@]dbuniversity.ac.in

Abstract: *There are several applications that can be related to multimedia content analysis. Considering video as one of the prominent forms of multimedia content, this paper presents analysis of human walking motion (gait) found in video sequences by using promising strategy of integrating techniques from data fusion and computer vision. To provide solutions to the challenges in human gait analysis a unified framework is proposed comprising of three different levels: data level, feature descriptor level and decision level. The three levels perform specific tasks assigned to them. At the data level, features are extracted from input video sequences for minimal representation. At the feature descriptor level, features from minimal representation are rearranged to build a feature descriptor and finally at decision level meaningful interpretations are performed. For analyzing human walking motion found in video sequences, initially, moving silhouettes are extracted using background subtraction for minimal representation at the data level. The extracted silhouettes are then represented in a common representation in a spatial form followed by correlation analysis and a feature descriptor is developed with minimum interest points at the feature descriptor level. Finally, interpretation of normal gait poses and transition poses are made at the decision level.*

Keywords: *Multimedia content; Data Fusion; Unified Framework; Background Subtraction; Correlation; Feature Descriptor; interpretation of Gaits.*

1. Introduction

Multimedia is a term that collectively describes a variety of media content available in different combinations of text, speech, and audio, still images, video, animation, graphics, and 3D models. Much research studies were dedicated to performing automated computational tasks for a wide spectrum of applications such as surveillance, crime investigation and so on. Among all the available types of media, the video is one of the prominent forms, widely used for analyzing multimedia content and refers to the computerized understanding of the semantics of a multimedia document.

Computer vision is used to electronically perceive visual data in the form of images, text, videos. One of the main goals of researchers working in computer vision has been to enable computers to analyze and interpret images or videos. Processing of different kinds of images and videos in computer vision systems include several techniques and approaches for analyzing the content of multimedia to derive meaningful interpretations. The relation between multimedia content analysis and computer vision is a well-known challenge. There are several applications related to multimedia content analysis and computer vision however, a significant application chosen for multimedia content analysis in this paper is to automatically interpret human movement found in videos.

To provide solutions to the challenges of human movement analysis using videos, the paradigm of data fusion is recommended. Multimedia data fusion is a way to integrate multiple media, their associated features or integrate intermediate decisions to perform an analysis task. According

to B.V. Dasarathy, “Combining Multimedia data fusion is a formal framework in which are expressed means and tools for an alliance of data originating from different sources for the exploitation of their synergy in order to obtain information whose quality cannot be achieved otherwise” [1]. However, there are different types of multimedia fusion. Multimedia fusion is useful for several multimedia analysis tasks such as detection of humans, event identification, tracking of vehicles, human motion tracking and a wide range of applications.

In this paper, a theoretical framework called, the Unified framework is presented based on the level of abstractions at data, feature descriptor and decision levels for analysis of human gait found in video sequences. All the tasks associated with human gait analysis are aligned to the levels defined by the Unified framework.

The first level, i.e., Data level of the Unified framework, the task was minimal representation. For this purpose, the initial task of human gait analysis, a method that is simple and effective for silhouette extraction is developed. A variety of algorithms experimented for detecting and tracking of humans walking found in videos and the proposed silhouette extraction method, Recurrent Blockwise GMM Background modelling on RGB colour space with tracking (RBGMM) is found to be effective in reducing the size of the storage with minimal representation.

The second level, i.e., Feature descriptor level, tasks such as choosing a common representation, correlation analysis and building of a feature descriptor were defined. For the



silhouettes extracted in the previous data level, a common representation in a spatial form, known as the common coordinate system was used followed by correlation analysis. Next, a feature descriptor was developed to integrated two kinds of features, PbHOG and 2D-SIFT for minimum interest points that remained stable over all actions or poses of gait.

The Third Level, i.e., Decision level, The interpretation of predicting normal gait poses is achieved by Supervised classification using Multi-class linear Support Vector Machine (SVM) and semantic interpretation of noticing changes of poses during walking, features were analysed using Single hidden layer feed forward neural network training algorithm using dynamic Extreme Learning Machine (ELM) algorithm.

The following section of this paper is organized as follows. Section 2, a literature review on various classification categories of data fusion, techniques of data fusion – correlation and prediction, an overview of human motion analysis with respect methodologies, major tasks of human motion analysis - human detection, human motion tracking, feature extraction and pose analysis of human walking (gait analysis). It also reviews selected benchmark datasets available and concludes with remarks. Section 3, provides details of proposed Tri-level Unified Framework and proposed human gait analysis tasks aligned to the framework. Section 4, presents details of experiments, results and analysis of the Tri-level Unified Framework and human gait analysis tasks aligned to the framework. Section 5 concludes with a summary and conclusion.

2. Related Works

The focus this research paper, is to employ data fusion classification framework to human gait analysis using computer vision techniques. This section presents data fusion frameworks and techniques and, state-of-the-art research work on human gait analysis, an overview of the bench mark datasets and followed by a discussion on the relationship of data fusion strategies to human movement analysis.

2.1 Data Fusion Frameworks and Techniques

Categories of data fusion different types of data fusion classifications are categorized as:

1. JDL data fusion classification where data fusion levels are defined by the JDL as level 0, 1, 2, 3, 4 [1].
2. Dasarathy's classification systems based on the input/output data types and their nature, as proposed by Dasarathy [2].
3. Classification based on abstraction levels of data such as fusion- early, late and hybrid [3].

The data fusion techniques are classified as correlation, prediction and decision fusion. Correlation refers to establishing a set of measurements observed over time, prediction refers determination of the state or position by measurements defined by correlation and decision fusion refers to deriving higher level of meaning. Correlation is often

performed before prediction or decision fusion and is applicable to all levels of fusion.

2.2 Overview of Human Gait Analysis

In any type of human motion analysis applications, the initial step is to detect human. Commonly used techniques are background subtraction[4], temporal differencing technique [5] and optical flow technique [6]. After human detection, Human motion tracking, is carried out, where the corresponding human figures detected are tracked that depict temporal correspondences. Some widely used tracking techniques are Kalman filter[7], Dynamic Bayesian Network [8], Condensation algorithm [9]. Following this, feature extraction is performed to specify some quantifiable property that is significant for computations. Generally, the features include, color, texture and shape [10].

Pose analysis of human walking (gait analysis) is the final task in accomplishing an understanding of human walking motion. The pose analysis can be processed in two steps namely, pose estimation and action recognition. The pose analysis can be accomplished by supervised or unsupervised learning. T. B. Moeslund[11] separated pose estimation algorithms as model-free, indirect model and model-based approaches. Actions were defined by the contexts of applications by Nagel [12] as hierarchy of change, event, verb, episode and history. Bobbick [13] defined as different levels of abstraction – movement, activity and action.

3. Proposed Methodology

From the literature review, it is found that there is need for developing a strategy to analyse human gait in videos (multimedia content). Therefore, this section describes the proposed methodology which is developed and implemented in two-step approach. Namely,

- a. Design of Tri-level Unified Framework and
- b. Analysis of human gaits.

3.1 Design of Tri-level Unified Framework

The visual content found in the video sequences is modelled as a hierarchy of abstractions at the three levels namely, 1. Data level, 2. Feature Descriptor level and 3. Decision level.

In the data fusion of multimedia content we fuse together the information.

1. Data level – At this level the different modalities can be combined and compression can be made to reduce the size of the storage.
2. Feature Descriptor level – At this level, several processing tasks such as spatial and temporal alignments, feature extraction, semantic and radiometric alignments can be made.
3. Decision level- this level includes the common representation and decision labelling

For example, to analyse and interpret human walking present in a video sequence, the Data level extracts silhouettes for minimal representation, i.e., reduce the size of the storage. The

Feature Descriptor chooses silhouettes and rearranges them into a common representation and builds a feature descriptor. The Decision level uses the feature descriptor to analyse for a meaningful interpretation of walking.

A brief description of the unified framework is given in fig 1.

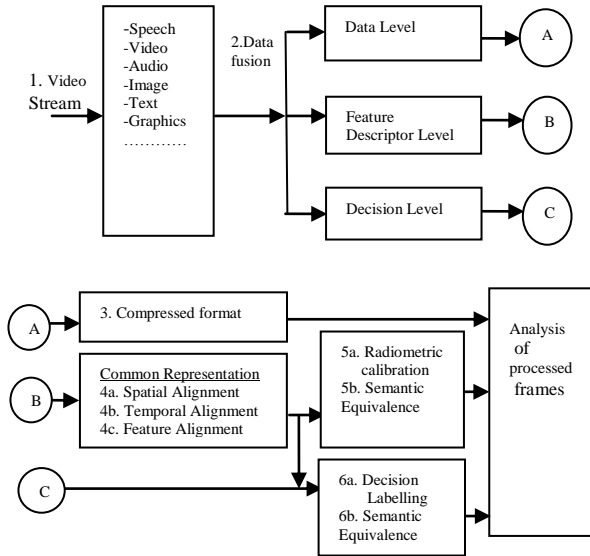


Figure1: Proposed Tri-level Unified Framework

1. *Video Stream*: Input of video sequences
2. *Data Fusion*: In the data fusion we fuse together the information contained in the multiple modes at various levels depending on the complexity of the data at - data level, feature level and/or decision level.
 - a. *Data level*- At this level, the different forms can be combined and compression can be done to reduce the size of the storage. After the fusion further processing can be done for multimedia analysis.
 - b. *Feature level*- At this level, a common representation can be presented so that the input types of media “speak a common language”. This involves several processing including: spatial, temporal, feature extraction, semantic and radiometric alignment.
 - c. *Decision level*- This level includes the common representation and decision labelling
3. *Compressed Format*: The direct input of video stream is large in size hence various techniques can be used for reducing the storage size by defining a minimal representation scheme.
4. *Common Representation*: The principal functions in the common representation are:
 - a. *Spatial Alignment*-The input frames/segments are spatially aligned with the same geometric base. Without a common geometric base any information derived from a given input

image cannot be associated with other spatial information. The accurate spatial alignment of the input images/ sequence of frames is therefore a necessary condition for multimedia fusion.

- b. *Temporal Alignment*-The spatially aligned input images/ scenes/ frames/audio segments are temporally aligned to a common time. This step is only required if the input images/ scenes/audio are changing or evolving in time. In this case the accurate temporal alignment is a necessary condition for multiple data fusion.
 - c. *Feature Extraction*-Characteristic features are extracted from the spatially and temporally aligned input images/ frames. The output is one or more feature maps for each input image/ frame/segment of audio.
5. a) *Radiometric Calibration*: The spatially, temporally and semantically aligned input images/ videos/text and feature maps are converted to a common measurement scale. This process is known as radiometric calibration.
 6. a) *Decision labelling*: Pixels in each spatially and temporally aligned input image/frame or feature map are labelled according to a given criteria. The output is a set of decision maps.
 - 5b) & 6b) *Semantic Equivalence*: In order for the input images/ frames/segments of audio, feature maps or decision maps to be fused together they must refer to the same object or phenomena. The process of causally linking the different inputs to a common object or phenomena is known as semantic equivalence[14].

3.2 Analysis of Human Gaits

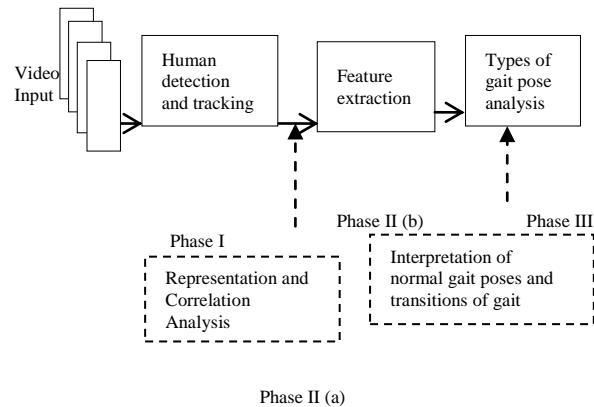


Figure2: Proposed Human gait analysis

The human gait analysis is carried in three phases that are outlined in Figure 2, as, 1. Human detection and tracking, 2. Feature extraction and 3. Types of gait pose analysis. Initially, a simple and effective method is developed to extract the moving silhouettes. Features are rearranged by aligned representation in one dimensional vector followed by the correlation analysis and finally interpretation of poses as normal and interpretation of transitions are performed. The

tasks of gait analysis are aligned to the hierarchy of abstractions as proposed by the unified framework.

1. *Phase I - Human Detection and Tracking*: During the task of detecting and segmenting humans found in video frames, Background subtraction method is used. The moving regions are detected by taking the difference between the incoming frames and the reference background modelled frame in a pixel wise fashion. The robustness of this technique depends mostly on the background model chosen to be as a reference frame. Generally, all background subtraction algorithms use two steps - 1. Background modeling and 2. Motion segmentation or foreground extraction. Many types of research have been reported, however, there is still need to improve and develop new effective approaches.

2. *Phase II (a) - Representation and Correlation Analysis*: While extracting silhouettes, the shapes of the silhouettes change over time. A common coordinate representation is used to represent the extracted silhouettes in a single dimension vector. Later, correlation analysis of the silhouettes across the frames is performed. The goal of correlation analysis in this work is to establish a set of observations or measurements that are generated by the same detected human silhouette over time and analyse the overall performance.

3. *Phase II (b) - Feature Extraction*: Feature descriptor is one which provides a set of optimized features extracted from the previous stage of silhouette representation. Feature extraction is a process of describing a set of useful measurements of the extracted silhouettes that can be used for the further process in analysis tasks. There are several types of silhouette-based features.

4. *Phase III - Types of Gait Pose Analysis*: The final step in the human gait analysis is to feed the feature extracted attributes computed in previous stages into a classifier to infer desired knowledge about the gait pose. Two types of tasks are experimented using neural networks.

Type 1: Interpretation of normal gait poses - walk, run, jump, bend using SVM classifier and

Type 2: Interpretation of transitions – automatic detection of a transition of poses during walking using feed forward networks of ANN.

The proposed Tri-level Unified Framework is used as a baseline approach for dividing the tasks associated with the human gait analysis into three levels – Data level, Feature Descriptor level and Decision level.

4. Experimental Datasets

4.1 Dataset # 1: Weizmann Dataset [15]

The dataset contains 93 low-resolution (180 x 144, with speed rate 50 fps) video samples for human actions each video consists of 70 – 110 frames. The video format used for experiments is in '.avi' format. There are 10 different actions in this dataset such as bending, jumping, walking and one hand waving, jumping in place, running, gallop sideways, skip jumping and two hands waving. These actions were performed by 9 actors.



Figure 3: Sample frames of Weizmann dataset

4.2 Dataset #2: i3DPostMulti-view Human Action Dataset [16]

The database has been created using eight convergent cameras setup to produce high definition multi-view videos. Various types of motions are recorded, where each video depicts one of eight performing one of the twelve different human motions. The database consists of 832 single view videos- each of 120 to 125 frames. The 12 different actions consist of the walk, run, jump forward, jump in place, bend, one hand wave, sit down - stand up, walk – sit down, run – fall, run – jump – walk, two persons handshaking, one person pulls another. Figure 4 shows a sample of actions found in the i3DPost Multi-view Human action dataset.

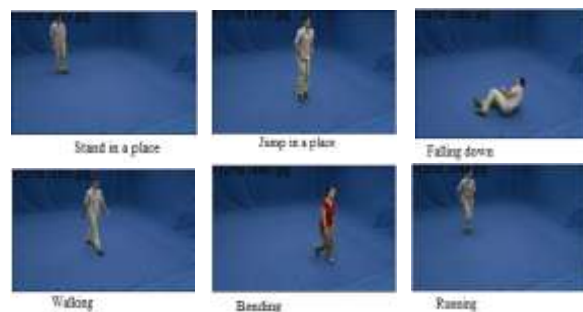


Figure 4: Sample frames of i3DPost Multi-view Human action dataset

4.3 Dataset # 3: Own Cross Dataset

A video was compiled of images of different actions. 50 images of various actions of walking, running, sitting and dancing. Each frame consisted of a different action and different background, with a resolution (364 x 288). The video played 15 fps for each action.



Figure 5: Sample frames of own cross dataset

5. Experiments, Results and Analysis



This section describes the analysis of human gaits using the Tri-level Unified framework. Silhouette extraction is used for minimal representation relating to the Data level. Common representation scheme and correlation analysis are performed and feature descriptors are built at the Feature Descriptor level. At the Decision level, using the features from Feature Descriptor level, analysis of human gaits as i) detection of normal gait poses – walking, running, jumping and bending ii) interpretation of transitions are performed.

The overall proposed work is implemented in two stages - Training stage and Testing stage.

1. Training stage:

The training stage is used to train program about the different types of gait poses including normal as well as transitional poses. The training stage always starts before testing stage in human action analysis. It consists of several processes: reading a training video, computing the common coordinate representation of silhouettes, computing features based the extracted silhouettes, preparing feature vector and saving the feature vector for further processing tasks. All these steps are repeated for each sample of training videos found in the datasets chosen. All processes of training stage are repeated for all training samples in the dataset. At the end of training stage, the feature vectors are made available.

The major steps of Training stage:

Step 1: Input of videos required for training

Step 2: Pre-processing of frames

Step 3: Detection of moving humans

Step 4: Represent extracted silhouettes in common coordinate representation and perform correlation analysis

Step 5: Build feature descriptor

2. Testing stage:

The testing stage is a program that interprets the normal gait poses happening in the video and also detects transitions, i.e., changes of poses from one pose to another. The testing stage consists of several processes such as reading the testing video, computing the feature vector for the testing video, followed by action classification based on training feature vectors and identifying action happening in the testing video. Until this point, these steps are common to the steps of training stage and have to be exactly in the same manner in every detail, for the comparison of the classification algorithm is performed successfully. Later the process of classification feature vector for testing video is applied using one of action classification methods. The proposed classifier algorithms are SVM and single hidden layer feed forward neural networks.

The training mode starts by reading training videos and classifies them into respective classes. The testing mode starts by reading the query video and later the normal gait poses and

transitions are matched to their respective classes and thus the ends the process, after all, the video sequences are processed.

The major steps in Testing stage:

Step 1: Input of Query video

Step 2: Pre-processing of frames

Step 3: Detection of moving humans

Step 4: Represent extracted silhouettes in common coordinate representation and Perform correlation analysis

Step 5: Build feature descriptor

Step 6: Interpretation of normal gaits and interpretation of transitions

5.1 Experiment I: First level – Data Level of Tri-level Unified Framework

In most applications relating to human gait analysis, the initial step is to detect moving humans. Segmentation is used to obtain relevant information about the human figures located in the frames of human motion. The methods presented in this section have been focused on one of the segmentation methods known as background subtraction method.

Existing motion segmentation algorithms were extended, and a simple, effective algorithm is proposed for obtaining clean silhouettes. The various methods investigated include:

1. Adaptive background modeling using OTSU threshold method.
2. Simple background modeling on ‘V’ layer in HSV color space followed by morphological operations.
3. GMM background modeling in HSV color space with tracking.
4. Blockwise GMM background modeling on RGB color space without tracking.
5. Recurrent blockwise GMM background modeling on RGB color space with tracking. (Proposed algorithm)

5.1.1 Major Steps of Investigated Algorithms

- a. Adaptive Background Modeling using OTSU Threshold Method

Step 1: Preprocessing: Normalize all frames

$$r' = r/255, g' = g/255, b' = b/255 \quad (1)$$

Where r, g, b represent the red, green, blue color layers and r', g', b' are the normalized.

Step 2: Differencing of frames: Extract motion pixels by computing the difference between the incoming current frame and background model.

$$Diff_{t+1} = |I_t - B_{t-1}| \quad (2)$$

Where I_t is the pixel value of current frame and B_{t-1} is the pixel value of the background estimate. The background pixel is assumed to be a single Gaussian (μ, σ).

Step 3: Modeling: Initially obtain single Gaussian of all pixels of the first frame and use it as the reference frame and update the next consecutive frames as

$$B_{t-1} = \alpha I_t + (1 - \alpha)B_{t-1} \quad (3)$$

Where, α is the learning speed of the background. (The value of α is chosen to be 0.5)

Step 4: The differenced image is threshold by OTSU [17], is an adaptive threshold that is obtained recursively by taking the variance of the Gaussian model.

b. Simple Background Modeling on ‘V’ layer in HSV Color Space followed by Morphological Operations

Step 1: Preprocessing: Normalize all frames

$$r' = r/255, g' = g/255, b' = b/255 \quad (\text{same as 1})$$

Where r, g, b represent the red, green, blue color layers and r' , g' , b' are the normalized

Step 2: Convert RGB to HSV color space.

Step 3: Background modeling: Extract the ‘v’ layer and construct background model using one Gaussian (1-G) [18]. Model each pixel with a Gaussian distribution $\eta(\mu_{s,t}, \Sigma_{s,t})$. Where, $\mu_{s,t}$ average background color and $\Sigma_{s,t}$ Covariance matrix

Step 4: Silhouette extraction: calculate Mahalanobis distance between pixels of reference frame and the incoming frame Mahalanobis distance [19]:

$$d_M = |I_{s,t} - \mu_{s,t}| \Sigma_{s,t}^{-1} |I_{s,t} - \mu_{s,t}|^T \quad (4)$$

Where, $I_{s,t}$, $\mu_{s,t}$ are HSV vectors and $\Sigma_{s,t}$ is the covariance matrix.

Step 5: Compare the distance to a threshold value 30.

Step 6: Apply morphological operations - erosion and dilation to obtain better silhouettes

Step 7: Update mean and covariance matrix of each pixel of the background model

$$\mu_{s,t+1} = (1 - \alpha)\mu_{s,t} + \alpha I_{s,t} \quad (5)$$

$$\Sigma_{s,t+1} = (1 - \alpha)\Sigma_{s,t} + \alpha(I_{s,t} - \mu_{s,t})(I_{s,t} - \mu_{s,t})^T \quad (6)$$

c. GMM Background Modeling in HSV Color Space With Tracking

Step 1: Preprocessing: Normalize all frames

$$r' = r/255, g' = g/255, b' = b/255 \quad (\text{same as 1})$$

Where r, g, b represent the red, green, blue color layers and r' , g' , b' are the normalized

Step 2: Convert RGB to HSV color space.

Step 3: Background modeling: Construct background model for every pixel using Stauffer and Grimson [20] mixture of k Gaussians. Thus the probability of occurrence of a color at a given pixel is given by:

$$P(I_{s,t}) = \sum \omega_{i,s,t} \cdot \eta(I_{s,t}, \mu_{i,s,t}, \Sigma_{i,s,t}) \quad (7)$$

Where $\eta(I_{s,t}, \mu_{i,s,t}, \Sigma_{i,s,t})$ is the i^{th} Gaussian model and $\omega_{i,s,t}$ its weight. The covariance matrix $\Sigma_{i,s,t}$ is diagonal. (Usually $I_{s,t}$ is within 2.5 standard deviation of its mean)

Step 4: Parameters of matched component are updated as follows:

$$\omega_{i,s,t} = (1 - \alpha)\omega_{i,s,t-1} + \alpha \quad (8)$$

$$\mu_{i,s,t} = (1 - \rho)\mu_{i,s,t-1} + \rho \cdot I_{i,s,t} \quad (9)$$

$$\sigma_{i,s,t}^2 = (1 - \rho)\sigma_{i,s,t-1}^2 + \rho(I_{i,s,t} - \mu_{i,s,t})^2 \quad (10)$$

Where α is the learning rate given by user usually taken as 0.0005 and ρ is the second learning rate given as $\rho = \alpha\eta(I_{s,t}, \mu_{i,s,t}, \Sigma_{i,s,t})$

Step 5: Once the Gaussian has been updated, the K distributions are normalized so they sum up to 1. The K distributions are ordered based on fitness value $\omega_{i,s,t}\sigma_{i,s,t}$ and only the H most reliable are chosen as part of the background

$$H = \text{argmin}_h (\sum \omega_i > T) \quad (11)$$

Where, T is the threshold. Those pixels which are more than 2.5 standard deviations away from H distributions are labeled “in motion” and are tracked by bounding box.

d. Blockwise GMM Background modeling on RGB Color Space Without Tracking

Step 1: Preprocessing: Normalize all frames

$$r' = r/255, g' = g/255, b' = b/255 \quad (\text{same as 1})$$

Where r, g, b represent the red, green, blue color layers and r' , g' , b' are the normalized

Step 2: Background modeling: Initially, on the first frame, construct background model using one Gaussian (1-G) and set it as the reference frame. Model each pixel with a Gaussian distribution $\eta(\mu_{s,t}, \Sigma_{s,t})$. Where, $\mu_{s,t}$ average background color and $\Sigma_{s,t}$ Covariance matrix

Step 3: Silhouette extraction: Divide the both the reference frame T1 and the current frame T2 into blocks (Block size = 8). Calculate Mahalanobis distance between pixels of reference frame and the incoming frame for all the blocks. Mahalanobis distance:

$$d_M = |I_{s,t} - \mu_{s,t}| \Sigma_{s,t}^{-1} |I_{s,t} - \mu_{s,t}|^T \quad (12)$$

Where, $I_{s,t}$, $\mu_{s,t}$ are RGB vectors and $\Sigma_{s,t}$ is the covariance matrix.

Step 4: $D = |T1 - T2|$ (13) Step 5: Compare D to a threshold given by

$$Th = (\max((\max(D)))) / 1.3 \quad (14)$$

(with 50% tolerance)

Step 6: Select all the moving and non-moving pixels in the current frame and set non-moving to 0 and moving to 1

Step 7: Select the updated current for new mean and covariance matrix of each pixel of the background model

$$\mu_{s,t+1} = (1 - \alpha)\mu_{s,t} + \alpha I_{s,t} \quad (15)$$

$$\Sigma_{s,t+1} = (1 - \alpha)\Sigma_{s,t} + \alpha(I_{s,t} - \mu_{s,t})(I_{s,t} - \mu_{s,t})^T \quad (16)$$

e. Recurrent Blockwise GMM Background Modeling on RGB Color Space With Tracking (Proposed algorithm)

Step 1: Preprocessing: Normalize all frames

$$r' = r/255, g' = g/255, b' = b/255 \quad (\text{same as 1})$$

Where r, g, b represent the red, green, blue color layers and r', g', b' are the normalized

Step 2: Background modeling: Initially, on the first frame, construct background model using one Gaussian (1-G) and set it as the reference frame. Model each pixel with a Gaussian distribution $\eta(\mu_{s,t}, \Sigma_{s,t})$. Where, $\mu_{s,t}$ average background color and $\Sigma_{s,t}$ Covariance matrix

Step 3: Silhouette extraction: Divide the both the reference frame T1 and the current frame T2 into blocks (Block size = 8). Calculate Mahalanobis distance between pixels of reference frame and the incoming frame for all the blocks. Mahalanobis distance:

$$d_M = |I_{s,t} - \mu_{s,t}| \Sigma_{s,t}^{-1} |I_{s,t} - \mu_{s,t}|^T \quad (\text{same as 12})$$

Where, $I_{s,t}$, $\mu_{s,t}$ are RGB vectors and $\Sigma_{s,t}$ is the covariance matrix.

Step 4: $D = |T1 - T2|$ (same as 13)

Step 5: Compare D to a threshold given by

$$Th = (\max((\max(D)))) / 1.3 \quad (\text{same as 14})$$

(with 50% tolerance)

Step 6: Select all the moving and non-moving pixels in the current frame and

set non-moving to 0 and moving to 1

Step 7: To update T2 for next iteration, each iteration compares D to the similarity of at least 30 matches in the next five consecutive frames. Select only those frames with > 30 matches and set that frame as next current frame.

Step 8: Select the updated current for new mean and covariance matrix of each pixel of the background model

$$\mu_{s,t+1} = (1 - \alpha)\mu_{s,t} + \alpha I_{s,t} \quad (\text{same as 15})$$


$$\Sigma_{s,t+1} = (1 - \alpha)\Sigma_{s,t} + \alpha(I_{s,t} - \mu_{s,t})(I_{s,t} - \mu_{s,t})^T \quad (\text{same as 16})$$

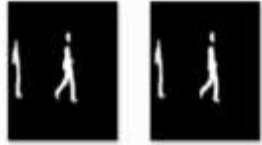



Step 9: After extraction of silhouettes all of them are aligned and tracked by their centroid coordinates.

5.1.2 Results of Extraction

The adaptive background modeling using OTSU threshold method yielded binarized difference images with many holes. The second method was implemented by converting the frames of video sequences from RGB to HSV, and the 'V' layer was extracted where the illumination variances were found to be stabilized. From the 'V' layer the silhouettes were extracted using GMM method and later refined with the morphological operations. The problem of pseudo silhouettes were present. The third method was implemented using GMM in HSV color space for human detection and tracking. This method was not useful for the low resolution video sequences of Weizmann however for the high resolution images of i3DPost Multi-view Human action dataset, humans were detected and were tracked with bounding boxes the silhouettes were not binary silhouettes. The fourth method was implemented by dividing the all the RGB frames in the video sequence into blocks. By applying single Gaussian background modeling along with threshold, moving silhouettes was extracted, and the resultant silhouettes were noisy. The fifth method is a proposed method that improvised upon the previous method by intuitively selecting the next frame which is less similar to the current frame and recurrently all the selected frames were processed with noise threshold and finally, the silhouettes with least noise were extracted and the proposed method is found to be the desired the best on both low as well as high resolution video sequences.

TABLE 1: TYPES OF SILHOUETTES EXTRACTED USING VARIOUS METHODS SPECIFICATIONS

Methods Investigated	Types of Silhouettes
1. Adaptive background modeling using OTSU threshold method.	

<p>2. Simple background modeling on 'V' layer in HSV color space followed by morphological operations.</p>	
<p>3. GMM background modeling in HSV color space with tracking</p>	
<p>4. Blockwise GMM background modeling on RGB color space without tracking</p>	
<p>5. Recurrent blockwise GMM background modeling on RGB color space with tracking. (Proposed algorithm)</p>	

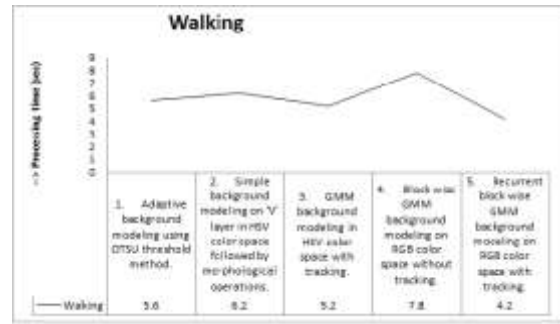


Figure 7: Processing time comparison for walking

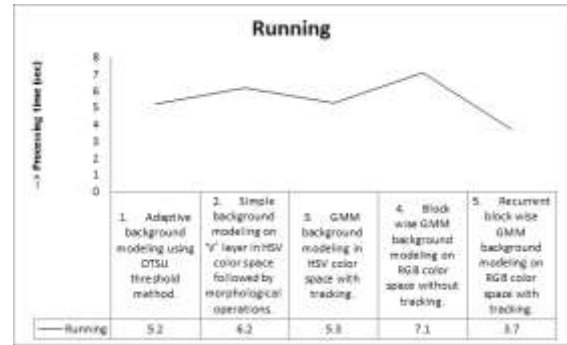


Figure 8: Processing time comparison for running

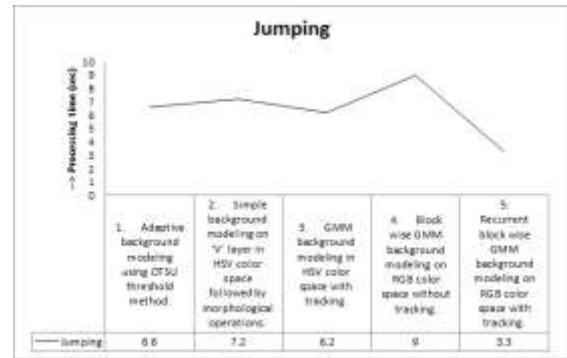


Figure 9: Processing time comparison for jumping

5.1.2 Analysis on Results of Extraction

The investigated methods discussed in the previous section were evaluated and compared in terms of speed of execution and noise levels found in Weizmann and i3DPost multi-view human action datasets.

The processing time was calculated for different types of gait - walking, jumping, running, bending using the investigated methods. It is noticed that the proposed method Recurrent Blockwise GMM Background Modeling and Tracking is fast in the extraction of silhouettes.

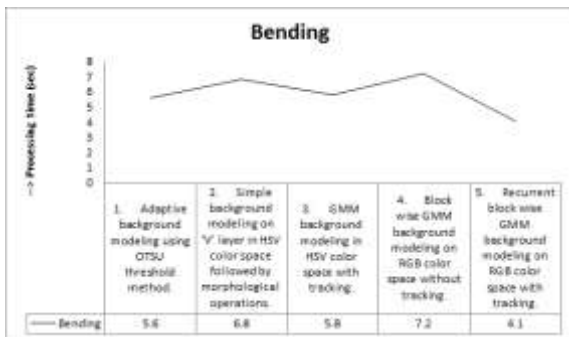


Figure 6: Processing time comparison for bending

The percentage of noise levels were qualitatively compared by considering the silhouettes generated by the proposed algorithm, Recurrent Blockwise GMM Background modeling and tracking developed as the probe silhouettes.

TABLE II: PERCENTAGE OF NOISE LEVEL PRESENT IN THE SILHOUETTES EXTRACTED USING VARIOUS METHODS

Methods/Actions	Walk-ing (%)	Jump-ing (%)	Run-ning (%)	Bend-ing (%)
Running average with OTSU	40	45	44	55
HSV segmentation with morphological operations	35	40	38	45
GMM in HSV	30	20	25	15

color space with tracking				
Blockwise GMM Background modeling	25	32	25	50
Recurrent Blockwise GMM Background modeling and tracking	2	4	1	15

5.2 Experiment II: Second level– Feature Descriptor level of Tri-level Unified Framework

A common representation scheme, correlation analysis and extraction of features for building feature descriptor are tasks associated with the Feature Descriptor level of the Tri-level Unified Framework. For human gait analysis, common representation scheme, known as Aligned Representation for silhouettes extracted from the previous level, is followed by correlation analysis for reduced processing time and overheads. A feature descriptor is built by integration of silhouette and contour based features.

5.2.1 Representation of Silhouette Shapes

The representation of silhouette shapes was implemented with the extraction of boundary points using radii and angular distance along the outer contour points of the silhouettes. The centroid of the human blob is determined using the equations:

$$X_c = 1/N \sum X_i \tag{17}$$

$$Y_c = 1/N \sum Y_i \tag{18}$$

(Xc, Yc) is the average contour pixel position and (Xi, Yi) represent the points on the contour of the blob. The radii distance is calculated as

$$D_i = \sqrt{(X_c - X_i)^2 + (Y_c - Y_i)^2} \tag{19}$$

and the angular distance is calculated as

$$\theta = \tan^{-1} (Y_c - Y_i) / (X_c - X_i) \tag{20}$$

Each shape in anticlockwise is unwrapped into a set of boundary points with details of radii distance and angular distance sampled along the outer contour in a common coordinate representation. Each gait sequence is accordingly converted to an associated sequence of such 2D-shape configuration. Since the boundary trace is developed using the displacements between the points and aligned to a common coordinate representation, it is invariant to translation but not to the rotation. So, any variance in the rotation is considered to be as one of the gait poses.

Once the spatial silhouette of walking is extracted, the boundary points are obtained, and the centroid positions along with angular distances of all silhouettes are computed and are

represented in a vector of single dimension as shown in the Fig. 10.



Figure 10: Selected silhouettes represented in a single dimension vector

5.2.2 Results of Silhouette Correlation Analysis Based on Human Identification

One of the common techniques, of data fusion, is data association or correlation which looks at the goal of data fusion to obtain a lower detection error probability and a higher reliability of data. The impact of correlation analysis was compared for Data overhead and Time complexity

TABLE III: OBSERVATIONS OF CORRELATED INFORMATION OF THE TWO METHODS BGMM AND RBGMM FOR RUNNING

OBSERVATIONS BGMM	OBSERVATIONS RBGMM
Original Sample Size = 478720	Original Sample Size = 478720
Redundant coefficients = 457581	Redundant coefficients = 464260
Motion Elements detected = 21139	Motion Elements detected = 14460
Data overhead = 4.4157%	Data overhead = 3.0206%

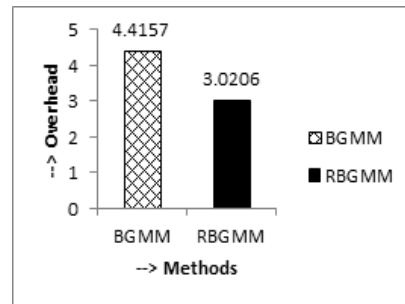


Figure 11: Data overhead comparison

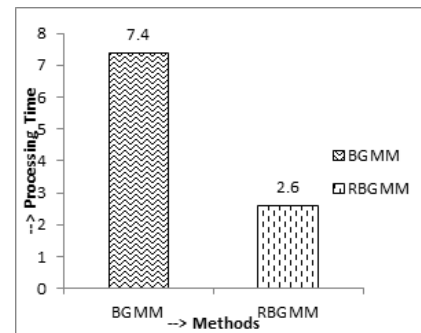


Figure 12: Time complexity comparison

Table III shows the observations of the original pixel count of the silhouettes detected during running. Redundant information



is calculated by counting all zero elements, and the motion elements are calculated by subtracting the redundant information from the original pixel count. The overhead is calculated by dividing motion elements detected by the original pixel size of the detected silhouette. It is noticed that the overhead of the proposed Recurrent Blockwise GMM (RBGMM) method is less than that of Blockwise GMM (BGMM) method. Similarly, other poses also demonstrated the same

5.2.2 Algorithms Investigated for Feature Extraction

Following algorithms were investigated for building feature descriptor.

1. Histogram of gradient (HOG)
2. Probability histogram of gradient (PbHOG)
3. Scale Invariant feature Transform (2D-SIFT)
4. Optimised feature extraction using PbHOG and optimized 2D-SIFT (proposed algorithm)

a. Histogram of Gradient (HOG)

The method is based on evaluating well-normalized local histograms of image gradient orientations in a dense grid. In practice, this is implemented by dividing the image window into small spatial regions (“cells”), for each cell computations of the HOG are based on magnitudes and angles of the gradients [5] [21]. This feature is extracted from the image based on two parameters: number of overlapping cells (N x N) and the number of bins (B) for the gradient angles. The gradients of an image are computed by filtering image with horizontal kernel [-1, 0, 1] and vertical kernel [-1, 0, 1]¹. Then magnitudes and angles are computed based on computed gradients. For each cell, angles are binned into B orientations based on their angles’ values. For each bin sum of gradient, magnitudes are calculated. After which, the sums which are equal to the number of bins are normalized. In the end, N x N x B normalized values are obtained and stored in a feature vector. These are called as HOG feature descriptors.

b. Probability Histogram of Gradient (PbHOG)

This section gives an overview of the probability of boundary (Pb) method [22] and the PbHOG method. The task is to use features extracted to estimate the posterior probability of a boundary conditioned on some image features and then extract HOG features based on the Pb responses. The advantage of the probability of boundary (Pb) method is Pb responses suppress more noises. The Pb method analyses the image features based on optimized cues of intensity, brightness, color and texture combined to obtain boundary strength. The brightness feature considers two computations - Oriented energy, brightness gradient, the color feature considers the color gradient and the texture gradient computes the texture gradient. In order to compute a feature vector components for a pixel include a circular patch around the pixel. The patch is divided into two halves at an angle θ horizontal. Eight orientations and three scales σ to compute features are used. Based on this pixel patch, Following are the mathematical equations used for

computing the cues. In the CIE L*a*b* color space (luminance, red-green, yellow-blue)

Brightness features.

- i) Oriented Energy

$$OE_{\theta,\sigma} = (I * F_{\theta,\sigma}^e)^2 + (I * F_{\theta,\sigma}^o)^2 \quad (21)$$

f^e : Gaussian second derivative

f^o : Its Hilbert transform

- ii) Brightness gradient BG (x,y,r, θ)

The chi-square difference in L* distribution of the two halves of the circular patch is one which results in the brightness gradient.

Color feature.

Again in the CIELAB space, Color gradient CG (x,y,r, θ) is Chi-square difference in a* and b* distributions (joint or marginal). a and b terms correspond to the perceptually orthogonal red-green and yellow-blue color components.

Texture feature.

Texture gradient TG(x,y,r, θ) is Chi-square difference of texton histograms, which is computed by performing clustering resolving using k-means by giving a unique identifier known as texton to the center of each cluster of histograms. Textons are vector-quantized filter output.

The HOG features extracted are considerably large and usually include noise, therefore, to enhance feature set extraction, the probability of boundary (Pb) operator which has been shown to perform well in delineating the detected human silhouettes is used and then based on the Pb responses, HOG features are extracted.

c. Scale Invariant feature Transform (2D-SIFT)

The 2D SIFT [23] feature set is achieved using a scale-space kernel function such as Gaussian, which is a continuous function that captures stable features in different scales. The Gaussian function on two points x and y can be defined as:

$$G(x, y, \sigma) = 1/2\pi\sigma^2 (e^{-\frac{(x^2+y^2)}{2\sigma^2}}) \quad (22)$$

While the scale-space function L(x, y, σ) that define the input image I (x, y, σ) can be defined as:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y, \sigma) \quad (23)$$

Where * is the convolution operation.

After that, defining stable locations in the scale space D (x, y, σ) can be achieved by the convolution of an input image with the difference of the Gaussian (DoG) functions:

$$D(x, y, \sigma) = (G(x, y, K\sigma) - G(x, y, \sigma)) * I(x, y, \sigma) \quad ..(24)$$

$$= L(x, y, K\sigma) - L(x, y, \sigma) \quad ..(25)$$

Where, the DoG function is the difference between two neighboring scales with a K constant distance.

Finally, defining the maxima and the minima of $D(x, y, \sigma)$ gives the scale invariant points in the scale-space.

d. Optimised Feature Extraction using PbHOG and Optimized 2D-SIFT (proposed algorithm)

In order to reduce the features probability of boundary (Pb) was introduced, however, interest based approach of SIFT have the advantage of short feature vectors. Therefore, in this work, an enhanced way by combining PbHOG with SIFT is experimented and the optimization is performed on the interest points by fine tuning in the form of introducing a threshold value on the number interest points detected and extracted. Initially, the threshold value is given to be 5.

Major steps in Interest based integrated PbHOG and 2D-SIFT

Step 1: Extract features using PbHOG method

Step 2: Extract features using 2D-SIFT

Step 3: Calculate the number of interest points = I_p , and compare I_p to a threshold, Th

If $I_p > 30$ then $Th = 15$

If $I_p > 20$ then $Th = 12$

If $I_p > 10$ then $Th = 10$

Else $Th = 5$

The threshold here determines a number of details the features can actually be used for the next level of work. The idea of using threshold for the features extracted is to reduce the weak interest points and concentrate only on those which can actually help to perform next level of the task by reducing the overheads.

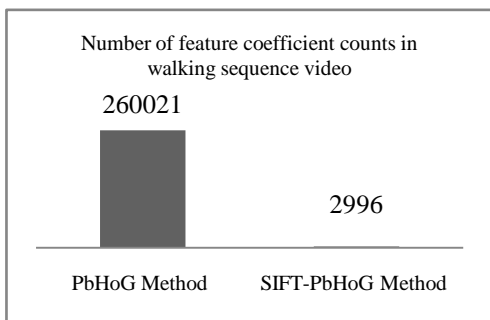


Figure 13: Size of reduced Optimized feature set using integration of PbHOG and 2D-SIFT

5.3 Experiment III: Third level – Decision level of Tri-level Unified Framework

This section elaborates on the tasks related to decision level of the Unified framework. The analysis of human gait poses is categorized as

- i) Detection of normal gait poses- walking, running, jumping and bending using supervised multi-class linear SVM classifier, and

- ii) Interpretation of transitions - automatic interpretation of transitions in poses of human walking using feed forward networks of ANN.

Along with Dataset # 1, Dataset # 2 and Dataset #3 were used for experimentation.

5.3.1 Experiment #1: Using Own Cross Dataset and Multiclass Linear SVM Classification

The experiment was conducted using a video compiled of images with 50 different actions. The video played 15 fps for each action during the training phase of the dataset in the Multiclass Linear SVM; the PbHOG feature extraction is used to extract features from each of the image present in the frame. The training instance matrix is formed from the feature values extracted and provided to the Multiclass Linear SVM Classifier for training. In the testing phase, the test data set is in the form of a video which comprises of images for various actions. The video is first split into frames, and the extracted PbHOG features are provided to the recognizer. SVM predictor then classifies the test data into different actions. The various actions are classified as walking, running, sitting and dancing.

To evaluate the retrieval efficiency of the approach using Multiclass Linear SVM Classifier, the performance measures of precision and accuracy are calculated. The accuracy and the precision factor are defined as:

$$\text{Recall or Accuracy} = \frac{\text{(number of relevant pose images retrieved)}}{\text{(total number of relevant pose images present)}} \times 100 \quad \dots(26)$$

$$\text{Precision} = \frac{\text{(number of relevant pose images retrieved)}}{\text{(number of pose images retrieved)}} \times 100 \quad \dots (27)$$

5.3.2 Experiment#2: Using Weizmann dataset and Multiclass Linear SVM Classification

The experiment for interpreting the normal type of gait was performed by using the features extracted from PbHOG and SIFT on Weizmann dataset. The combined feature set was fed to the Multiclass SVM classifier. For a classification problem of predicting normal gait types, with ‘M’ classes. The four classes defined for the implementation included class 1 as walking, class 2 as running, class 3 as jumping and class 4 as bending.

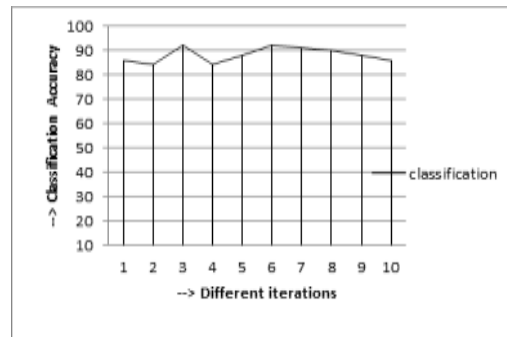


Figure 14: Classification accuracy using SVM classifier

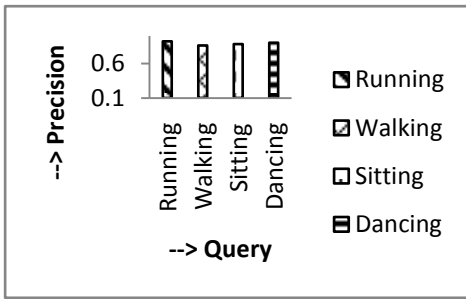


Figure 15: Classes of human action taken for experimentation

Four SVMs are trained each using examples of one class as positive training example and the examples of the other as negative training examples. To classify a query example, all the four SVMs are evaluated, and the class label of the SVM with the largest value of the decision functions is selected.

To evaluate the retrieval efficiency of the approach using Multiclass Linear SVM Classifier, the performance measures of precision and accuracy are calculated. The accuracy and the precision factor are defined as:

$$\text{Recall or Accuracy} = \frac{\text{(number of relevant pose images retrieved)}}{\text{(total number of relevant pose images present)}} \times 100 \dots\dots\dots \text{(same as 26)}$$

$$\text{Precision} = \frac{\text{(number of relevant pose images retrieved)}}{\text{(number of pose images retrieved)}} \times 100 \dots\dots\dots \text{(same as 27)}$$

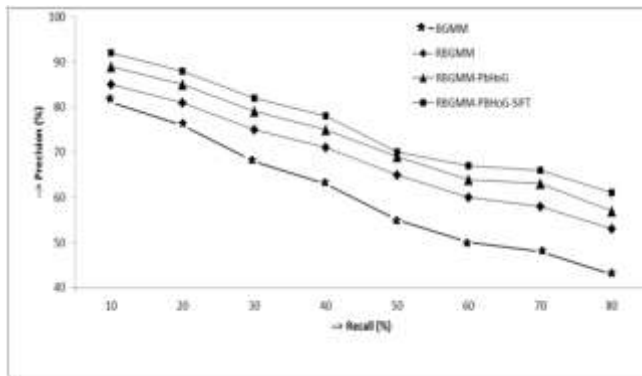


Figure 16: Precision and Recall values of different methods used for experimentation

The Fig. 16 shows a graph depicting the precision and recall values of the various methods used for predicting the normal gait. The Fig. 16 shows the curves of the moving human detection using the methods

1. Blockwise GMM background modeling on RGB color space without tracking (BGMM) and
2. Recurrent Blockwise GMM background modeling on RGB color space with tracking (RBGMM).

The Fig. 16 also shows the classification of types of gait using

1. Extracted features of PbHOG from the silhouettes extracted using BGMM and

2. Extracted features of PbHOG and SIFT from the silhouettes extracted using RBGMM.

The curves show that the performance of classification, of the proposed Recurrent Blockwise GMM background modeling on RGB color space with tracking (RBGMM) method used for extraction of silhouettes and the features extracted using Interest based Integrated PbHOG and SIFT to be better than the features extracted using only PbHOG for classification of the gait types.

A variety of summary statistics can be computed. Each statistic highlights the recognition system’s performance relative to different criteria. Most commonly used metrics Accuracy and Precision are presented below

From the Table IV, spatio-temporal volume method has better accuracy than our proposed Recurrent Blockwise GMM background modeling on RGB color space with tracking (RBGMM) method used for extraction of silhouettes and the features extracted using Interest based Integrated PbHOG and SIFT. But since the 3D view is taken into consideration the spatio-temporal volume method suffers from large memory requirement.

Table IV: COMPARATIVE ANALYSIS OF PROPOSED METHOD AND OTHER METHODS

Methods	Localized Multiple Kernel Learning	Hierarchical Filtered Motion	Spatio Temporal Volume	PbHOG + SIFT+ Neural network
Precision (%)	43.14	95.7	77	90
Accuracy (%)	77.91	93.6	89.2	73.46

It is also observed that the Hierarchical Filtered Motion method has higher precision and accuracy, but the drawback is it works well only with a moving and cluttered background. But the scope of this work is limited to the static camera. Hence, for a static background, our method gives the best accuracy and precision.

5.3.3 Experiment #3: Using i3DPost Human Action Dataset for Detecting Transitions Using Feed forward Networks

In, this section we propose an algorithm for Single hidden layer feed forward neural network training algorithm. The conventional Single hidden layer feed forward neural network training algorithms require the input weights and the hidden layer biases to be adjusted using a parameter optimization approach which usually leads to low performance and are confined to specific applications and cannot be generalized. Unlike the conventional style, Extreme Learning Machine (ELM) is an algorithm, that chooses input weights, and hidden layer biases randomly and the output weights are calculated [24].



In this work, we proposed an extension of ELM, called dynamic ELM, by looking at the mapping process of the input space where the dynamic grouping of training data is performed, and the output weights perform the semantic interpretation of setting a mark whenever there are changes in poses such as change of pose from walking to jumping, walking to sitting, walking to falling are noticed. The neural network could generate a series of values of each action and then compared with the action of each frame. The walking motion is considered to be normal, and all other type of gaits is considered to be abnormal based on the values categorized in the experiment performed. The proposed neural network is used for supervised classification.

The dynamic ELM network is a network that consists of D input (equal to the dimensionality of x_i), L hidden and C output (equal to the number of classes involved in the classification problem) neurons. The number of hidden layer neurons is usually selected to be much greater than the number of classes, i.e., $L \gg C$.

Let $\{x_i, c_i\}$, $i = 1 \dots N$ be a set of N vectors x_i belonging to D followed by class labels c_i belonging to $\{1 \dots C\}$. The network target vectors $t_i = [t_{i1} \dots t_{iC}]^T$, each corresponding to a training vector x_i , are set to $t_{ik} = 1$ for vectors belonging to class k, i.e., when $c_i = k$, and to $t_{ik} = -1$ otherwise. The network input weights W_{in} are arranged using Active Lezi scheme belonging to $R^{D \times L}$ and the hidden layer bias values b belonging to R^L are randomly assigned while the network output weights W_{out} belonging to $R^{L \times C}$ are analytically calculated. For a given activation function of the network hidden layer $\Phi(\bullet)$ and linear activation function of the network output layer, the output $o_i = [o_{i1} \dots o_{iC}]^T$ of the network corresponding to x_i is calculated by the following equation:

$$o_{ik} = \sum_{j=1}^L W_{kj} \Phi(V_j \cdot b_j \cdot x_i) \quad (28)$$

Where, W_{ki} is the j^{th} element of W_k , V_j is the j^{th} column of W_{in} and W_k is the k^{th} row of W_{out} .

Several activation functions $\Phi(\bullet)$ can be used for the calculation of the network hidden layer outputs, like sigmoid, sine, Gaussian, Radial basis Functions (RBF). In our experiments, we have used Gaussian function and normalization of values. The activation function is a matrix Φ , and the output neurons can be expressed as $W_{out}^T \Phi$. The input weights of the poses for training are calculated using simple prediction that can be performed to show the upcoming activities based on earlier observed sequences. A prediction algorithm based on the LZ78 text which performs compression [14] is useful to get a simple prediction for the correlated sequences of features. LZ78 processes an input string of characters, which in our case is a set of features.

The prediction algorithm parses the input string x_1, x_2, \dots, x_i into $c(i)$ substrings, or phrases, $w_1, w_2, \dots, w_{c(i)}$ such that for all $j > 0$, the prefix of the substring w_j (i.e., all but the last character of w_j) is equal to some w_i for $1 < i < j$. Because of the prefix property used by the algorithm, parsed substrings can be maintained in a trie along with frequency information.

The determination of network output weights are calculated, and a test vector x_i is introduced to the trained network and the all the variations are marked to note the changes in the gait when present.

Semantic interpretation of sudden transition of poses while walking was experimented on i3DPostMulti-view human action datasets of which three video sequences consisted recordings of transitions of normal running to falling down, walking to sitting and running to walking. The Dynamic learning of transitions was learned using the single hidden layer Feedforward neural networks using the dynamic ELM network, where the input neurons computed the input weights using the LZ78 prediction algorithm and the Gaussian activation on the hidden layer produced output neurons of several inputs. The test video notices the variations and whenever the changes in gait are noticed they are marked.

The standard classification error types that are used to calculate the metrics and build the performance visualizations are the two classes that are often called the positive and negative class leading to the familiar four possible results: true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN).

The Table IV, gives the number of false positives, false negatives that are obtained using our proposed method on three selected video sequences. % Percentage of (False Positive or False Negative) = Number of frames (False Positive or False Negative) / Total number of frames * 100 (29)

Table IV: FALSE POSITIVES AND FALSE NEGATIVES FOR VIDEO SEQUENCES 1, 2, 3

	Video Stream			
	Sequence 1	Sequence 2	Sequence 3	Total
False Positives	10 9.6%	15 12.09%	12 9.6%	9.94%
False Negatives	3 2.4%	5 4.03%	3 2.4%	2.95%

6. Summary and Conclusion

In this paper, the work presented has sufficiently demonstrated how the representation, analysis of multimedia content can be aligned to data fusion abstraction levels of the Unified

Framework. However, with respect to the application of analysis and interpreting of human gaits, implemented using abstraction levels of the Unified Framework, the scope of the present work can be extended in three main aspects. 1. Application perspective, 2. Increase in a number of cameras and 3. Extend to a high semantic analysis such as predicting the intent of a pose.

References

- [1] B. Khaleghi, A. Khamis, F. O. Karray and S. N. Razavi, "Multisensor data fusion: A review of the state-of-the-art." *Information Fusion* 14, vol. 1, pp. 28-44, 2013.
- [2] B. V. Dasarathy, "Sensor fusion potential exploitation-innovative architectures and illustrative applications." *Proceedings of the IEEE* 85, no. 1, pp. 24-38, 1997.
- [3] R. C. Luo, C.-C. Yih, and K. L. Su, "Multisensor fusion and integration: approaches, applications, and future research directions," *IEEE Sensors Journal*, vol. 2, no. 2, pp. 107-119, 2002.
- [4] K. P. Karmann and A. Brandt, "Moving object recognition using an adaptive background memory," in: V. Cappellini (Ed.), *time-Varying Image Processing and Moving Object Recognition*, vol. 2, Elsevier, Amsterdam, The Netherlands, 1990.
- [5] J. M. Chaquet, E. J. Carmona and A. Fernandez-Caballere, "A Survey of video datasets for human action and activity recognition", *Computer Vision and Image Understanding*, vol. 117, pp. 633-659, 2013.
- [6] C. W. Chu, O. C. Jenkins and M. J. Mataric, "Marker less Kinematic Model and Motion Capture from Volume Sequences", *Proceedings of IEEE Computer Vision and Pattern Recognition*, Wisconsin, USA, 2003.
- [7] G. Welch and G. Bishop, "An introduction to the kalman filter." *University of North Carolina: Chapel Hill, North Carolina, US*, 2006.
- [8] V. Pavlović, J. M. Rehg, T. J. Cham and K. P. Murphy, "A dynamic Bayesian network approach to figure tracking using learned dynamic models." In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 1, pp. 94-101. IEEE, 1999.
- [9] M. Isard, Michael and A. Blake, "Condensation—conditional density propagation for visual tracking." *International journal of computer vision* 29, no. 1, pp. 5-28, 1998.
- [10] R. S. Choras, "Image feature extraction techniques and their applications for CBIR and biometrics systems." *International journal of biology and biomedical engineering* 1, no. 1, pp. 6-16, 2007.
- [11] T. B. Moeslund, A. Hilton and V. Krüger, "A survey of advances in vision-based human motion capture and analysis." *Computer vision and image understanding* 104, no. 2, pp. 90-126, 2006.
- [12] H. H. Nagel, "From image sequences towards conceptual descriptions." *Image and vision computing* 6, no. 2, pp. 59-74, 1988.
- [13] A. F. Bobick, "Movement, activity and action: the role of knowledge in the perception of motion." *Philosophical Transactions of the Royal Society B: Biological Sciences* 352.1358, pp. 1257-1265, 1997.
- [14] S. N. Paul and Y. J. Singh. "Unified framework for representation, analysis of multimedia content for correlation and prediction." in *Emerging Trends and Applications in Computer Science (ICETACS)*, 2013 1st Int. Conference on, pp. 214-218. IEEE, 2013.
- [15] R. Auguste, A. E. Ghini, M. Bilasco, N. Ihaddadene and C. Djeraba, "Motion similarity measure between video sequences using multivariate time series modeling." In *Machine and Web Intelligence (ICMWI)*, 2010 International Conference on, pp. 292-296. IEEE, 2010.
- [16] N. Gkalelis, H. Kim, A. Hilton, N. Nikolaidis and I. Pitas, "The i3DPost multi-view and 3D human action/interaction," *Proc. CVMP*, pp. 159-168, 2009.
- [17] N. Otsu, "A Threshold Selection Method from Gray-Level Histograms," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, pp. 62-66, 1979.
- [18] Y. Benezeth, P.M. Jodoin, B. Emile, H. Laurent and C. Rosenberger, "Review and evaluation of commonly-implemented background subtraction algorithms." In *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, pp. 1-4. IEEE, 2008.
- [19] S. Xiang, F. Nie and C. Zhang, "Learning a Mahalanobis distance metric for data clustering and classification." *Pattern Recognition* 41, no. 12, pp. 3600-3612, 2008.
- [20] C. Stauffer and W. E. L. Grimson. "Adaptive background mixture models for real-time tracking." In *Computer Vision and Pattern Recognition, 1999. IEEE Computer Society Conference on*, vol. 2. IEEE, 1999.
- [21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection." In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1, pp. 886-893. IEEE, 2005.
- [22] D. R. Martin, C. C. Fowlkes and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues." *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 26, Vol. no. 5, pp. 530-549, 2004.
- [23] D. G. Lowe, "Distinctive image features from scale-invariant key points." *International journal of computer vision* 60, vol. no. 2, pp. 91-110, 2004.
- [24] A. Iosifidis, A. Tefas and I. Pitas, "Minimum variance extreme learning machine for human action recognition." In *Acoustics, Speech and Signal Processing (ICASSP)*, 2014 IEEE International Conference on, pp. 5427-5431. IEEE, 2014.

Author Profile



S. Nissi Paul, is a Research Scholar in the Dept of Computer Science & Engineering and IT, School of Technology, Assam Don Bosco University, Guwahati. She is pursuing her research in Artificial Intelligence and Computer Vision. She has completed M. Phil (Comp. Sc.) from Bharatidasan University in 2005.



Yumnam Jayanta, is working as Professor in the Dept of Computer Science & Engineering and IT, School of Technology, Assam Don Bosco University, Guwahati. He has received his PhD from Dr. B.A Marathwada University in 2004. He has worked with Swinburne University of Technology (AUS) at Malaysia campus, Misurata University, Keane (India and Canada), Skyline University College (UAE) etc. His research areas are ETL, Data Warehouse and Mining, Real-time Database system, and Image processing. He has produced several papers in International and National Journals and Conferences.