# Human Action Recognition System using Good Features and Multilayer Perceptron Network

Paper Id : 128

Int'l Conference on Communication and Signal Processing

Presented by :
Bhavana Mehta,
Jonti Talukdar .

Dept. of
Electronics and
Communication
Engg.

NIRMA UNIVERSITY
INSTITUTE OF TECHNOLOGY
NAAC ACCREDITED 'A' GRADE

# Human Action Recognition:

Characterize Human actions through automated analysis of video data.

## What are Human Actions?

**Actions in recent datasets:**

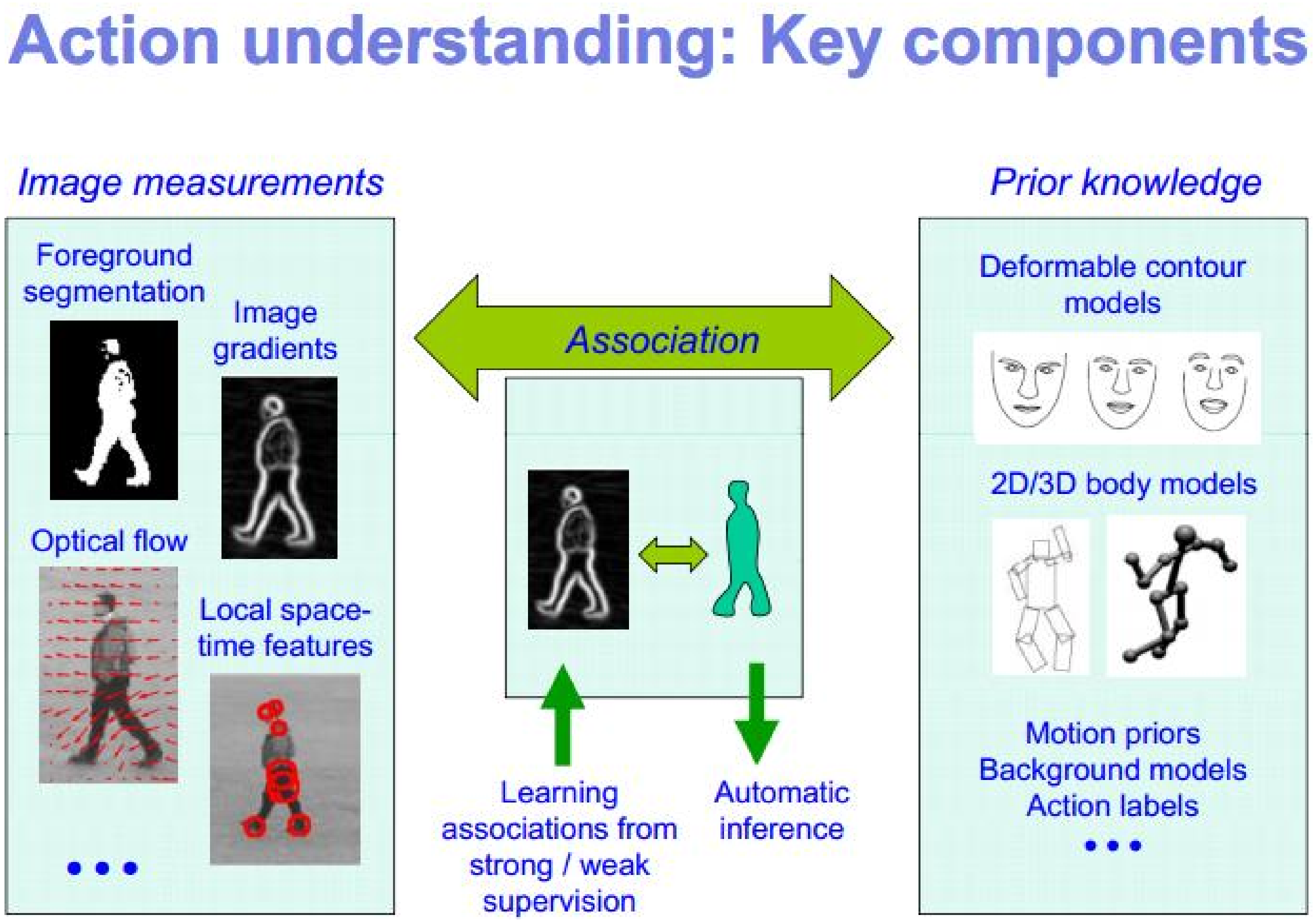
**Is it just about kinematics?**

**Should actions be associated with the function and *purpose*?**

**Kinematics + Objects + Scenes**

# Human Action Recognition:

Using standard video input to study, localize and analyse human actions using image features and machine learning..



# Existing HAR Methods:

Key Challenge: Reduce **Computational Complexity** while maintaining same level of **Accuracy.**

- Existing HAR broadly vary in the following domains:

1. **Feature Selection and Motion Representation:** Popular use of Local spatio-temporal feature representations of Human Actions [2-5],[6-8] versus Global representations.
2. **Training Methods:** Wide variety in classification techniques in use [10-13]. Neural Networks and deep learning are promising.

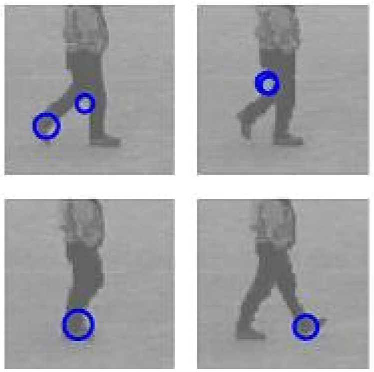


Fig. 1. Space time interest points [3].

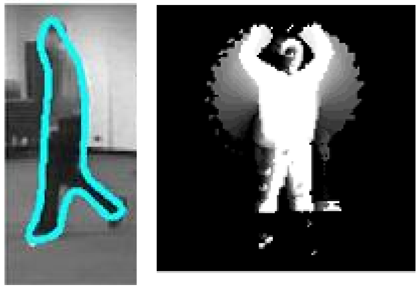


Fig. 2. Shape based features [6].

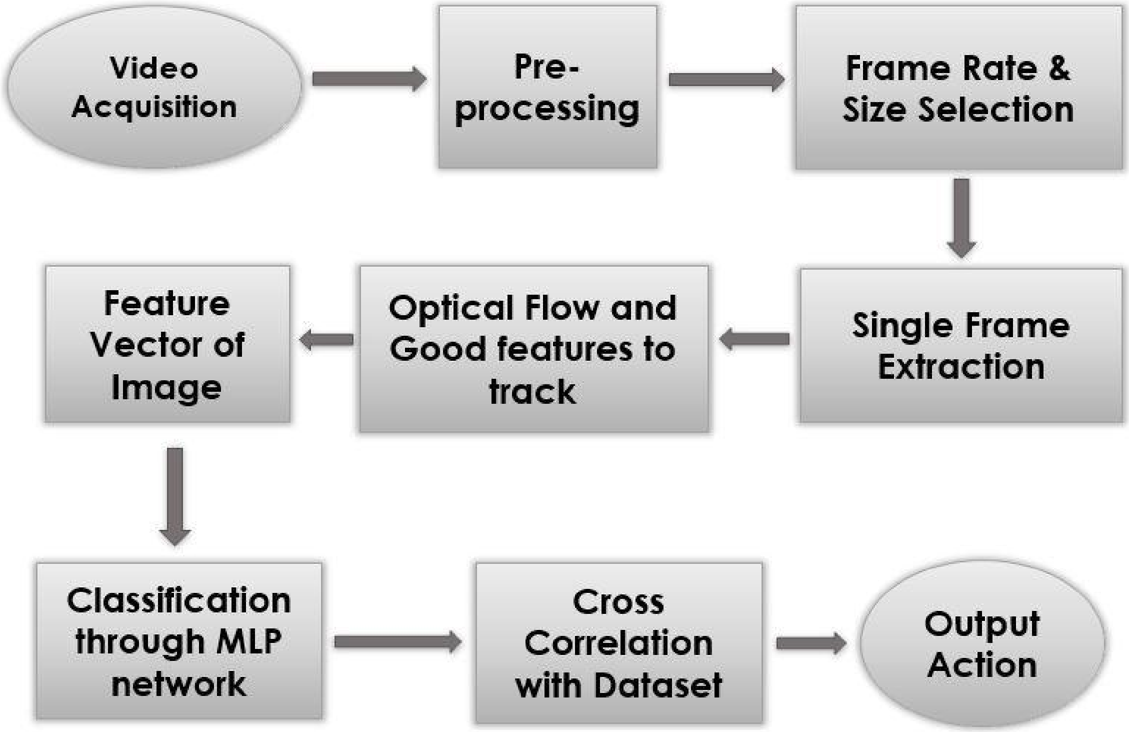
# Proposed HAR Method:

A combination of **Good Features** along with the **iterative optical flow** to achieve efficient feature extraction.

Use of **Multilayer Perceptron** neural network ensures efficient classification.

**Advantage:**
Computationally simple yet robust, produces good result.

## System Model for Human Action Recognition System

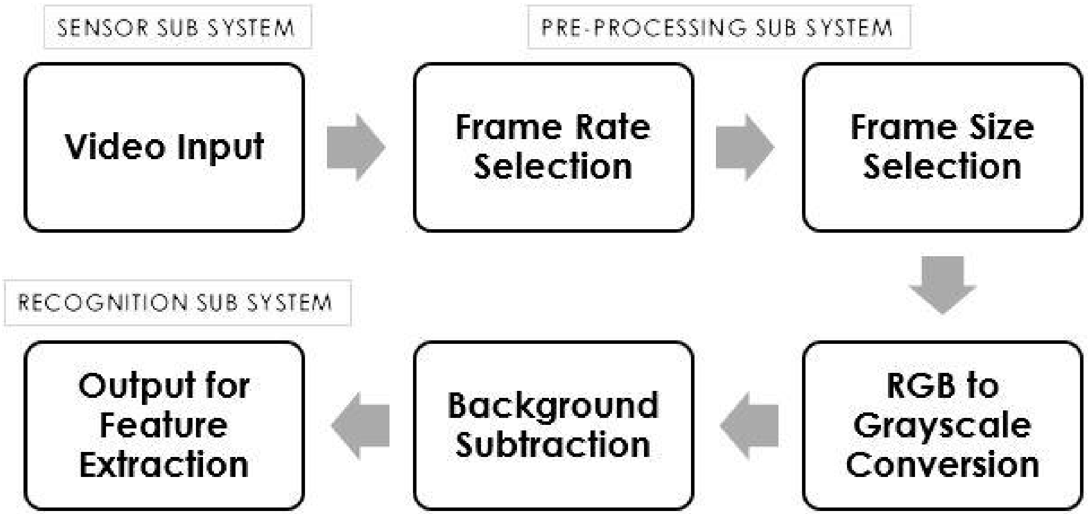


# Preprocessing Stage:

**1. Frame Rate and Size Selection:**
Input Video down sampled to 160*120 pixels.

**2. RGB to Grayscale Conversion:**
Grayscalling reduces size of dataset from 24 bits to 8 bits.

## System Architecture of the Preprocessing Stage




**3. Background Subtraction:** Gaussian mixture model used to model Probability of each pixel value 'x' at time N.

$$\Pr(x) = \sum_{k=1}^{K} w_k N(x; \mu_k, \varepsilon_k) \qquad (1)$$

**Fitness value $\boldsymbol{w_k/\varepsilon_k}$** gives measure of formation of static clusters.

# Feature Extraction Stage:

**Good Features & L-K iterative tracking algorithm used.**

N strongest good features tracked in video. LK algorithm applied to selected feature set to create an overall feature vector

Using motion based features with interest point features increases accuracy.

- An image sequence $I(x, y, t)$ under motion is represented as *:*

$$I(x, y, t+\tau) = I(x - \xi(x, y, t, \tau), y - \eta(x, y, t, \tau)) \qquad (2)$$

where $\boldsymbol{\delta} = (\xi, \eta)$ is displacement vector at point $\boldsymbol{X} = (x, y)$.

- Image motion between two frames can be represented as: $\delta = D\boldsymbol{X} + \boldsymbol{d}$

where D is the **deformation matrix** and $\boldsymbol{d}$ is the linear translation.

- The value of $\delta$ can be minimized by minimizing D and $\boldsymbol{d}$**,** which are in turn dependent on an error vector : $\boldsymbol{e} = Z\boldsymbol{d}$ .
- If $\lambda_1$ and $\lambda_2$ are the two eigenvalues of $Z$, then a good feature is selected only if: $min(\lambda_1, \lambda_2) > \lambda$ , where $\lambda$ is the tracking parameter.
- The feature vector $\boldsymbol{F}(x, y, t)$ obtained as follows:

$$\boldsymbol{F}(x, y, t) = [x, y, t, I_t, u, v, u_t, v_t, Div, Vor, Gten, Sten]^T$$

where $\boldsymbol{I}(\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{t})$ is the acquired image sequence, $u(x, y, t)$ is the corresponding optical-flow vector, $\boldsymbol{D_{iv}}$ is the spatial divergence of the vector field, $\boldsymbol{V_{or}}$ is the measure of local spin or vorticity of the flow fields, and $\boldsymbol{G_{ten}}$ and $\boldsymbol{S_{ten}}$ are invariant tensors.

# Classification Stage:

**MLP:** Multilayer Perceptron is a feed forward artificial neural network.

Efficiency improved by optimizing the number of layers as well as nodes per layer.

- Each neuron consists of a sigmoid activation function.
- Input feature vector undergoes nonlinear transformation downstream each node, given below.

$$y_i = \sum_j \left( w_{i,j}^{n+1} * x_j \right) + w_{i,j}^{n+1} + f(u_i) \qquad (3)$$

- Individual neuron weights adapted locally based on an error function, giving an update value $\Delta_{ij}$ for each $w_{ij}$.

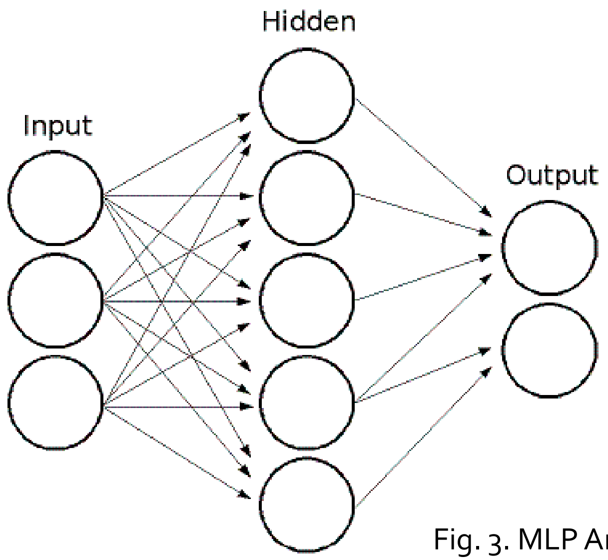



Fig. 3. MLP Architecture [12].

# Overall HAR Algorithm

**Algorithm:** Algorithm for HAR

**Input:** Video stream from static camera
**Output:** Recognized action class

1: Frame rate & size initialization = 160*120p.
2: RGB to Grayscale conversion.
3: **for each** Pr(x) at a given time $N$ **do**
4:     Evaluate fitness value $w_k/\varepsilon_k$.
5:     Subtract current frame and previous frame.
6: **end for**
7: **for each** Image sequence $I(x, y, t)$ **do**
8:     Evaluate deformation $D$ and linear translation $\boldsymbol{d}$.
9:     Initialize tracking parameter $\lambda$.
10:     **if** $min(\lambda_1, \lambda_2) > \lambda$ **then**
11:         Select feature for tracking.
12:         Evaluate feature vector $\boldsymbol{F}(x, y, t)$.
13:     **end if**
14: **end for**
15: Initialize number of feature vectors per frame, training samples, and number of hidden nodes in MLP.
16: Evaluate individual neuron weights $w_{ij}$ during training stages by passing training video data.
17: Pass feature vector $\boldsymbol{F}(x, y, t)$ to trained model for classification.
18: **return** recognized action class.

# Simulation Results:

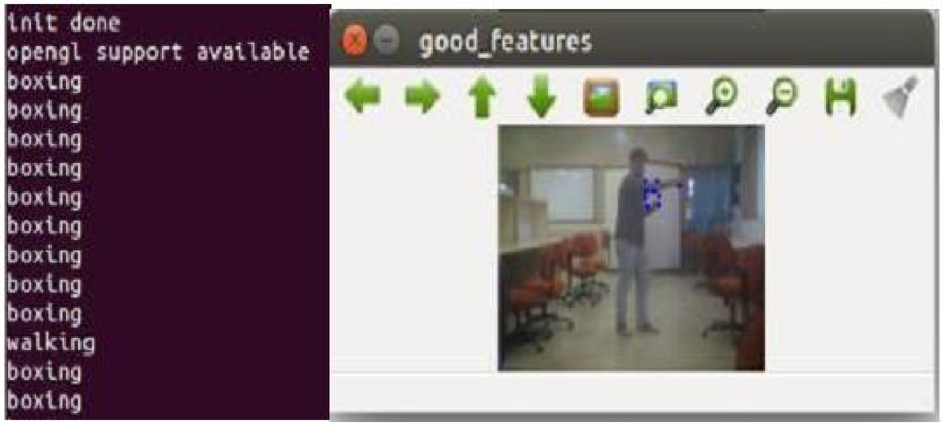




Fig. 4: Simulation Results implemented real time.

# Simulation Results:

## Recognition Rate of Four Action Classes

| Action Class | Recognition Rate | | |
|---|---|---|---|
| | Feature size 14 | Feature size 10 | Feature size 8 |
| Boxing | 95.2 | 93.2 | 89.8 |
| Clapping | 93.4 | 92 | 90.6 |
| Running | 95.2 | 94.3 | 89.4 |
| Walking | 94.6 | 93 | 88.4 |

- As the feature vector size increases, the recognition rate also increases.
- However, this relationship is nonlinear and there exists a point where increasing the feature size no longer improves the overall system performance.
- At this point, the lag or delay involved in processing the feature vectors outweighs the benefit in improved accuracy and the overall system performance reduces

# Conclusion

**Confusion Matrix for all Action Classes**

| | Boxing | Clapping | Running | Walking |
|---|---|---|---|---|
| Boxing | **112** | 7 | 2 | 0 |
| Clapping | 9 | **110** | 0 | 1 |
| Running | 1 | 0 | **113** | 6 |
| Walking | 2 | 0 | 7 | **111** |

- It can be observed that an overall system accuracy of **more than 92%** is obtained with running and boxing action classes having higher recognition rates.
- The final set of system parameters which can easily be implemented on a SBC consists of **200 hidden nodes** for the MLP, **a feature size of 10** and a training sample of **300 videos**.

# References


[1] C. Schuldt, I. Laptev and B. Caputo, "Recognizing human actions: a local SVM approach," Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004., 2004, pp. 32-36 Vol.3.

[2] I. Laptev and T. Lindeberg, "Space-time interest points," Proceedings Ninth IEEE International Conference on Computer Vision, Nice, France, 2003, pp. 432-439 vol.1.

[3] G. Chéron, I. Laptev and C. Schmid, "P-CNN: Pose-Based CNN Features for Action Recognition," 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, 2015, pp. 3218-3226.

[4] Weilong Yang, Yang Wang and G. Mori, "Human action recognition from a single clip per action," 2009 IEEE 12th International Conference on Computer Vision Workshops, ICCV Workshops, Kyoto, 2009, pp. 482-489.

[5] Jianbo Shi and C. Tomasi, "Good features to track," 1994 Proceedings of IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, 1994, pp. 593-600.

[6] K.Maithili, K. Rajeswari, R. Mohanapriya, D. Krithika, "An Efficient Human Action Recognition System Using Single Camera and Feature Points", International Journal of Advanced Research in Computer and Communication Engineering, Vol. 2, Issue 2, February 2013.

[7] R. Poppe, "A survey on vision-based human action recognition," Image Vis. Comput., vol. 28, no. 6, pp. 976–990, 2010.

[8] A. F. Bobick and J. W. Davis, "The recognition of human movement using temporal templates," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 3, pp. 257-267, Mar 2001.

[9] I. Laptev and T. Lindeberg. Local descriptors for spatiotemporal recognition. In Proceedings of ECCV Workshop on Spatial Coherence for Visual Motion Analysis, pages 91-103, 2004.

[10] J. C. Niebles, H. Wang, and L. Fei-Fei. Unsupervised learning of human action categories using spatial-temporal words. In BMVC, 2006.

[11] T. Hofmann, "Unsupervised Learning by Probabilistic Latent Semantic Analysis," Machine Learning, vol. 41, no. 2, pp. 177 196, 2001.

[12] D. Blei, A. Ng, and M. Jordan. Latent Dirichlet allocation. Journal of Machine Learning Research, 3:993–1022, 2003.

[13] Li, Stan Z., et al. "Statistical learning of multi-view face detection." European Conference on Computer Vision. Springer Berlin Heidelberg, 2002.

[14] J. Fan, W. Xu, Y. Wu and Y. Gong, "Human Tracking Using Convolutional Neural Networks," in IEEE Transactions on Neural Networks, vol. 21, no. 10, pp. 1610-1623, Oct. 2010.