Final Project: Activity Classification using MHI

Alon Amar

aamar32@gatech.edu

Recent Research

The first article [1] that this project is based on that laid the foundation, is the work of Prof. Bobick and Prof. Davis on recognition using temporal templates. In their work, they describe the ability to detect an action just from motion itself (without any 3D models). They used some earlier work and extended it to their use, mainly using Hu moments of the motion energy image (MEI) and motion history image (MHI) as described in our lecture. They use the covariance matrix and the mean of those actions as their features and use a Mahalanobis distance as a measurement to which action it most resembles to; they then classify a motion by that distance.

The second article [2] which the dataset is taken from is using a different technique to represent the motion in the video that is more stable with respect to changes of recording conditions. They use a local space time features which can be considered as primitive events corresponding to moving two-dimensional image structures at moments of non-constant motion – they are comprised of second-moment matrix using spatio-temporal image gradients and a histogram of Spatio-temporal neighborhoods of local features. They then use those features as an input to an SVM classifier they defined.

The third article [3] introduce the idea of dynamic image: “RGB still image that summarizes, in a compressed format, the gist of a whole video (sub)sequence” which by the article have several advantages over the older temporal frames like being able to be processed by CNN and efficiency. The dynamic image results then go into CNN, which process it similar to a still image.

My Implementation

My project is largely based on Bobick’s work, using temporal templates and calculating Hu moments and scale invariant moments as features. Those features are then used as an input to several machine learning algorithms in order to find the best result. The dataset from the second article is the input for this project.

My work can be summed up by the workflow in figure 1.

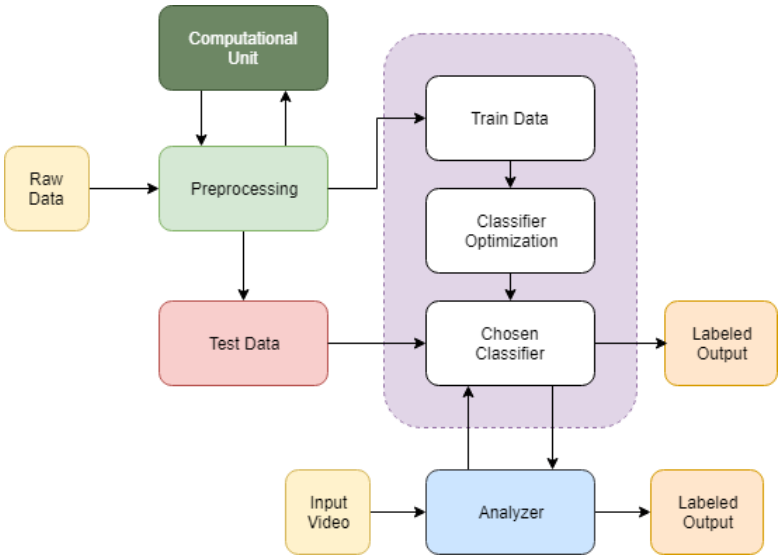


Figure 1: General workflow

Raw data

The raw data is the database provided to us including the frame boundaries for each action. This database contains sequences of six classes of actions performed by 25 subjects in four different conditions d1-d4. The database contains 2391 sequences.

http://www.nada.kth.se/cvap/actions/

Computational Unit

This unit consist of all the computation as described by the project instructions per image:

- Background subtraction – θ of 10 was chosen as it yield the best results.
- Motion History Image – τ is chosen as described in the preprocessing part.
- All types of moments – regular, central and scale invariant.

Preprocessing

- A background subtraction is calculated for each frame.
- The MHI is calculated per motion - while working on this project, I implemented two approaches:
    1. I started the preprocessing by computing the MHI for each frame boundary of each action. Meaning that the τ is different for each action and is set by the action boundaries (which are known in advance) to capture the full length of each motion. I call this approach ***Full Frame.***
    2. Since the first method also required prior information on any given future video, I implemented a second approach – setting a constant τ value (20) in order to capture the essence of each action. I found that τ=20, gives a distinguished MHI/MEI between each action (especially walking/jogging/running) and able to capture the full length of a motion like handwaving (from top to bottom). I call this approach simply ***Tau Frame.***
- The moments vector, which consists the Hu moments and the scale invariant moments of both MEI and MHI, is computed for each action.
- Each vector is also labeled as the class it belongs to:
  CATEGORIES = ["walking", "jogging", "running", "boxing", "handwaving", "handclapping"].

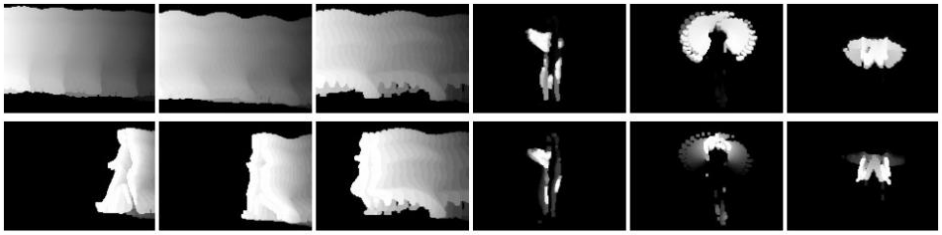


Figure 2: MHI for the 2 methods. Top row is the full frame approach, second row is the tau approach.

Train/Test Data

The given matrix provided in the preprocessing stage, is then divided into 2 portions. The training set consist of 75% of the total data, while the test is the remaining 25%.

Classifier optimization

Using the processed data, I started with 3 types of classifiers: K-Nearest Neighbors, Neural Network and SVM (all provided by the sklearn library). All are inspired by the 3 articles I described. By using cross-validation and a learning curve, I searched for the best parameters of each classifier. The best classifier is then selected to be the model classifier.

Analyzer

The analyzer is utilizing the trained classifier from the previous stage, to determine the motions in each provided video. We can look at it as the user application level – the user feed the video with the motion, and the analyzer provide an answer.

For each approach, the analyzer behaves differently:

1. Full frame – the frame boundaries for each motion are needed as a prior knowledge.
2. Tau frame – a sliding window of 20 frame is calculate each frame to determine the last action.

The analyzer produces a video with the labeled actions after each window of computation. In addition, it prints at the end the actions performed in the video.

Results

Classifier Optimization

Learning Curves

Since I Implemented 2 approaches (Full frame and tau frame) I will present the results for both. Since each classifier consists multiple parameters, I tuned the ones that seemed the most important to me, since tuning all of them has its computational difficulties.

- SVM: The 2 parameters I was looking for were gamma (Kernel coefficient) and C (Penalty parameter of the error term)
- KNN: The 2 parameters I was looking for were n_neighbors (Number of neighbors) and weights (weight function used in prediction).
- NN: The 3 parameters I was looking for were hidden_layer_sizes (number of nodes and hidden layers), solver (The solver for weight optimization) and alpha (L2 penalty).

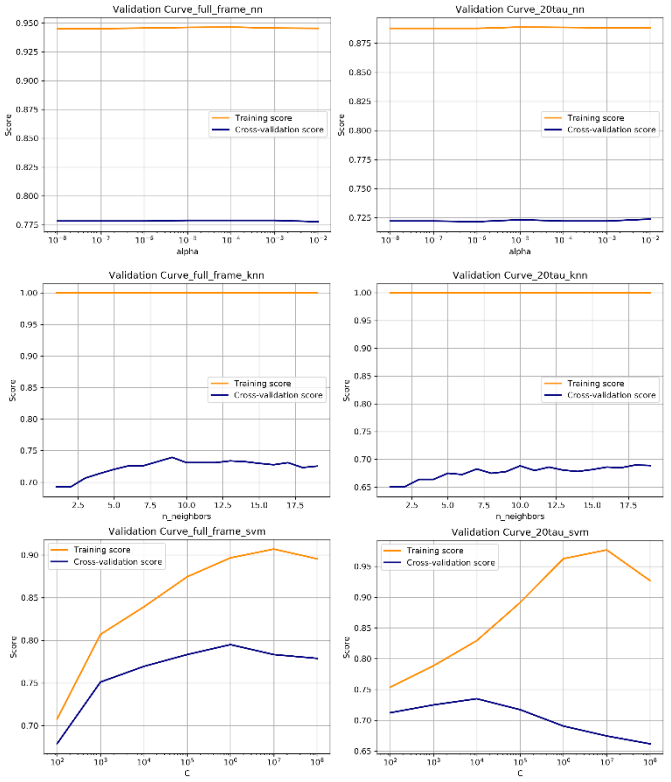
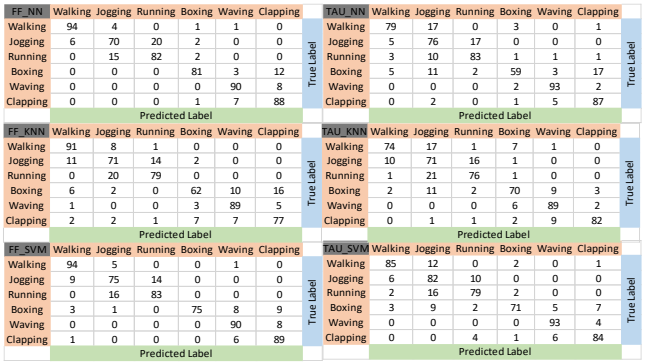


Figure 3: Validation curve for each classifier and method

We can see in figure 2 how the full frame approach is better than the tau frame, since we have more information about each motion, and we are capturing the full length of it. We can also see, especially in the SVM graphs, the optimum point for each classifier per parameter.



FF_NN	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	94	4	0	1	1	0
Jogging	6	70	20	2	0	0
Running	0	15	82	2	0	0
Boxing	0	0	0	81	3	12
Waving	0	0	0	0	90	8
Clapping	0	0	0	1	7	88

True Label / Predicted Label

TAU_NN	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	79	17	0	3	0	1
Jogging	5	76	17	0	0	0
Running	3	10	83	1	1	1
Boxing	5	11	2	59	3	17
Waving	0	0	0	2	93	2
Clapping	0	2	0	1	5	87

True Label / Predicted Label

FF_KNN	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	91	8	1	0	0	0
Jogging	11	71	14	2	0	0
Running	0	20	79	0	0	0
Boxing	6	2	0	62	10	16
Waving	1	0	0	3	89	5
Clapping	2	2	1	7	7	77

True Label / Predicted Label

TAU_KNN	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	74	17	1	7	1	0
Jogging	10	71	16	1	0	0
Running	1	21	76	1	0	0
Boxing	2	11	2	70	9	3
Waving	0	0	0	6	89	2
Clapping	0	1	1	2	9	82

True Label / Predicted Label

FF_SVM	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	94	5	0	0	1	0
Jogging	9	75	14	0	0	0
Running	0	16	83	0	0	0
Boxing	3	1	0	75	8	9
Waving	0	0	0	0	90	8
Clapping	1	0	0	0	6	89

True Label / Predicted Label

TAU_SVM	Walking	Jogging	Running	Boxing	Waving	Clapping
Walking	85	12	0	2	0	1
Jogging	6	82	10	0	0	0
Running	2	16	79	2	0	0
Boxing	3	9	2	71	5	7
Waving	0	0	0	0	93	4
Clapping	0	0	4	1	6	84

True Label / Predicted Label

Figure 4: confusion matrix for each classifier

Confusion Matrix and Scores

The confusion matrices for each classifier and method are shown in figure 3. We can see how across all classifiers there is a difficulty to differentiate between walking/jogging/running and boxing/waving/clapping since those actions have a similar motion which means a similar MEI/MHI.

The scores for each classifier on the test data, are shown in table 1.

	Full Frame	Tau Frame
SVM	86.20%	84.30%
KNN	79.89%	78.80%
NN	86%	81.30%

Table 1: Classifier results

I chose to take SVM for both methods.

Analyzer Results

There are 2 videos with multiple actions I fed the analyzer:

1. Action1 – includes walking, jogging and boxing.
2. Action2 – includes running, waving and clapping.

Both were stitched from the test set videos. Each type of action in repeated 4 times (as described in the dataset description, e.g. walking x 4, jogging x 4 etc.).

Table 2: Number of actions seen by each method

	Walking	Jogging	Running	Boxing	Waving	Clapping	
Tau Frame	63	105	15	56	14	1	Action1
	11	1	34	0	186	180	Action2
Full Frame	5	3	1	2	1	0	Action1
	1	0	4	0	4	3	Action2

We can see how the full frame recognition can indicate very nicely how much of each action has been performed. That is of course unfair, since he knows beforehand how many actions there are, and only need to classify them. The tau frame on the other hand, gives us the labeled action for each 20 frames window in the video with assurance level of 70% (using predict_proba of sklearn for that). On first look it might seem like it having a hard time detecting “running” for example, but “running” is a type of action that happens very quickly and will be seen less than walking or jogging for instance. An ideal classifier with 100% accuracy will output the number of frames each action has in the video (minus 20).

- The video output

Discussion

Why it's not perfect

Our features vector in its core is based on a simple background subtraction. That might cause some inaccurate data collection. In addition, MHI and MEI has their limitation as they can only described so much from a sequence of images. As described in the second article [2], this method can be fragile to changes of recording conditions. Another "issue" is our type of dataset – there are some actions that are relatively similar to one another and might be classified as the closed thing the classifier can find. A good example from the dataset, is "person22_boxing_d3_uncomp", which shows a man boxing while moving forward and the classifier result was jogging. We are also bound by computational power – analyzing each and every window of action in the training phase will take heavy computational power.

State of the art results

Since I took the dataset from the second article, I can compare my results to them. From their article they showed results of around 75% across different scenarios. For their training they used a smaller set of training data and used a different method for getting the temporal image information.

How can I improve

As described before, we can use more sophisticated method to get the motion detection, like median filtering that was described in the lecture. Another aspect is the size of tau. For my second approach, the value of tau as set after several observation. There might better way o determine the right tau for each action. Another perspective is using an entirely different motion detection model, as described in the third article. From the machine learning perspective, I used some basic classifiers that were provided by sklearn library. Tuning them even more or using CNN might yield better results.

Video Presentation

https://bluejeans.com/s/4IAMY/

https://www.youtube.com/watch?v=z5uLnVTAOk0&feature=youtu.be

Action Classification

Tau frame action1 - https://www.youtube.com/watch?v=wWlFUr-an4A&feature=youtu.be

Tau frame action2 - https://www.youtube.com/watch?v=WnVDtLO33Yc&feature=youtu.be

Full frame action1 - https://www.youtube.com/watch?v=eBA9KgtYttc&feature=youtu.be

Full frame action2 - https://www.youtube.com/watch?v=0ugY5cU40-A&feature=youtu.be

References

[1] James W. Davis, A. E. (1997). The Representation and Recognition of Human Movement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 257 - 267.

[2] Christian Sch¨uldt, I. L. (2004). Recognizing Human Actions: A Local SVM Approach.

[3] Hakan Bileny, B. F. (2016). Dynamic Image Networks for Action Recognition.