ISSN 2464-4617 (print)
ISSN 2464-4625 (CD)
Computer Science Research Notes
CSRN 2802
Short Papers Proceedings
http://www.WSCG.eu

Combination of Temporal Neural Networks for Improved Hand Gesture Classification

Aditya Tewari [‡][#]
Aditya.Tewari@dfki.de

Bertram Taetz[‡]
Bertram.Taetz@dfki.de

Frédéric Grandidier[†]
Frederic.Grandidier@iee.lu

Didier Stricker[*]
Didier.Stricker@dfki.de

ABSTRACT

Low latency detection of human-machine interactions is an important problem. This work proposes faster detection of gestures using a combination of temporal features learnt on block time input and those learnt by contextual information. The results are reported on a standard in-car hand gesture classification challenge dataset. The recurrent neural networks which learn sequential contexts are combined with 3D convolutional neural networks (C3D). We have demonstrated that a design similar to various multi-column networks, which have been successful for image classification and understanding can also improve classification performance on varying length time series. Therefore, a combination of C3D and Long-Short-Term Memory (LSTM) is utilized for classification of hand gestures. On the task of early hand gesture classification, the proposed model outperforms the the C3D model which reports best results on full gestures. It is second best and only slightly less accurate than the best performing method, on the full gesture length.

Keywords

LSTM, 3D Convolution, Neural Network, Temporal Features, Hand Gestures, Automobile Application.

1 INTRODUCTION

One of the principles for the interaction system is the need for a short time delay between the start of the interaction and response from the machine [RSP11]. A low latency system is easier to use and is often essential. Gesture recognition systems inside cars use sensors that require low power expenditure. These cameras introduce an integration time versus frame-rate trade-off [GYB04]. This reduces the available frame rate, thus a solution where robust predictions are made on short length gesture videos is important in such situations. Therefore, not only classification of complete fast gestures, but also classification of slow but incomplete gestures is of interest. In this work we solve the problem of robust and fast classification of (incomplete) hand gestures.

Several methods for vision based HGR have been employed. Hand crafted features [TTGS16] containing temporal and spatial information have been regularly used. Hidden Markov Models (HMM) [CFH03] and Support Vector Machines (SVM) [LGS08] have been used for classification of spatio-temporal features. Other solutions use an articulated model of the hand and its deformation for gesture classification [KKKA13]. It was empirically demonstrated by [AAGES10] that among machine learning methods, neural network models, like multi-layer-perceptrons (MLP), are conducive to early predictions in time series data. The Recurrent Neural Networks (RNNs) were used for gesture classification by [YH15]. The work [OBT14] and [WKSL13] reported results on the challenging VIVA dataset, the performance of these methods was overtaken by neural network based methods proposed in [MGKK15].

It has been demonstrated that multi-modal approaches [OBT14, WKSL13] which employ trajectory shapes, boundaries and motion structures combination in a bag of features approach work better than single information approach. The positive influence of mutually independent information contributing to learning have been demonstrated in such works. This approach has also shown to have worked with neural networks. A C3D network [MGKP15] was

[*] DFKI, Augmented Vision, Trippstadter Str., 67663 Kaiserslautern, Germany.
[†] IEE-SA, Weiergewan, 11-rue Edmond Reuter, 5326 Contern, Luxembourg.
[‡] TU Kaiserslautern, Erwin-Schrodinger-Strasse 1, 67663 Kaiserslautern,Germany.

trained with data from multiple vision sensors and radar for better hand gesture predictions. The neural network solutions that use Multi-information methods use parallel neural networks, these networks have performed better than single column networks and have been used in various image classification tasks [CMS12]. The performance of activity recognition [DAHG+15] and hand gesture recognition have shown to improve by using a combination of parallel networks that accepts distinct data [MGKK15, CLS15].

1.1 Contributions and structure of the paper

It is of interest to investigate if the combination of features learnt from the same dataset but using distinct learning policies, thus resulting in dissimilar patterns, can contribute towards improving the learning performance. This investigation is inspired from the improvement in performance of multi-modal network when data with separate properties is used at the various input layers. To improve classification over time by using dis-similar concepts, we create multi-column networks with columns constituted from different temporal layers. A typical model that we propose has a parallel columns with one or more volumetric convolution layer and one column with recurrent layers.
We introduce a hand gesture recognition system that uses a combination of C3D and LSTM for identifying gestures at different delays from the start of gestures. This work combines the ideas of [MGKK15] with those of [DAHG+15].
Results on the VIVA challenge dataset [viv], which is a hand gesture classification dataset recorded on varying lighting conditions inside a car, are demonstrated. On a half length, incomplete gesture sequences, our proposed network outperforms the two column C3D model by a large margin of approximately 10%. Improvement in performance is noticed and reported on short incomplete sequences of gestures. The combination model performs better on half and quarter length incomplete gesture sequence.
In this paper we propose a new neural network configurations that improves the classification performance for short sequence gestures. To the best of our knowledge, this is the first effort to utilize the combination of the two temporal neural networks to make early classification of a time series signal. Gesture sequences with full length (32 frames), half length (16 frames) and quarter length (8 frames) from the beginning of the gestures are trained and tested for these architectures. The contributions can be summarized as:

- Introducing a method for improving the accuracy of early response in a gesture recognition system.

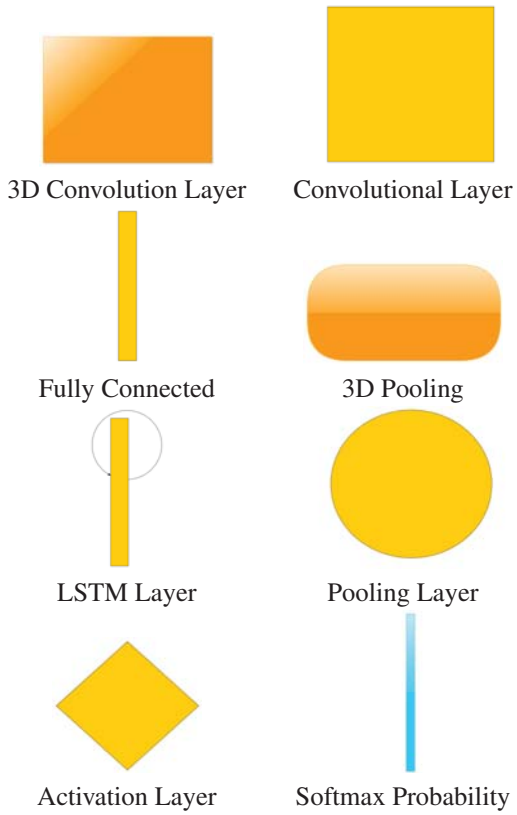

Figure 1: Labels for layer images used in this work

- Introducing a combination of block and context classification in time series by combining LSTM and C3D.

- An extensive analysis of various temporal neural network models for low latency classification of hand gestures.

In the Section 2 the dataset for the gesture experiments is explained, the sampling strategy and the preprocessing required to complete all the comparisons are described. The section Section 2.1 shows the training parameters and scheme used for various neural networks trained during the experiments. The Section 3 presents the C3D neural network that we later compare our proposed combination with. A benchmark is set in this section and test are also conducted with a multicolumn LSTM network. In the Section 4 we first train and validate the LSTM and C3D network and then propose a combination for better performance. These experiments are compared and the better performance of the combination network is demonstrated in the Section 6. The Section 7 presents an experiment on smaller models. Finally, in the Section 7.1, the possible extensions of this work and limitations are mentioned.

ISBN 978-80-86943-41-1

Figure 2: Representative Hand Gesture Dataset

2 DATA, SAMPLING AND TRAINING

The VIVA challenge dataset was used for these experiments. The gestures are defined by moving hands and changing or constant hand shapes. The VIVA challenge dataset has video sequences of fifteen hand gestures performed by eight subjects under varying lighting conditions inside a car. The dataset includes eight hundred and eighty-five intensity and depth video sequences [viv]. The dataset was recorded with the Microsoft Kinect device of resolution of 115×250 pixels and provides RGBD images.

The gesture length for each sequence in the gesture dataset is inconsistent. To create an equal length gesture, the normalization of the dataset sequence length is required. To compare with [MGKK15] the gestures were re-sampled to normal-length of thirty-two frames. If the length of a gesture sequence is less than the sequence it is reshaped into a normal-length sequence by up-sampling through repetition of frames. For sequences longer than the normal-length the gesture sequences were sub-sampled by dropping frames.

The normalized-length gesture sequences contain depth and intensity values. Intensity gradient values were calculated. The gradient and the depth values were normalized over the dataset and a two channel input from the gradient and normalized depth was created for each frame. The labels corresponding to the gesture type mark each frame. The gestures sub-sampling was done such that the the frame sequences with most variation in hand shape and motion were dropped with smaller probabilities. This is done by sampling based on magnitude of per pixel change over time within a gesture. The dense optical flow between two frames separated by time $\delta T = 2$ was calculated and the absolute change per pixel over the entire gesture was used for sampling distribution. This strategy allows improving the probability of conserving the fast changing frames during sub-sampling and increasing such frames when up-sampling the sequence.

Three classes which have 'Swiping' hand which changes direction in later parts of the gesture, and the class 'Tap three times' which is confused with the class 'Tap once' in early detection, were removed to analyze the performance. This was done because the these gestures are characteristically misidentified in short lengths. Effectively, the experiments were conducted on fifteen hand gesture classes.

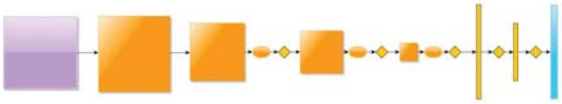

Figure 3: Single column, two input channel C3D.

2.0.1 *Short length sequences*

We focus on the improvement in latency performance for the time series so performance of classification on shorter gesture sequences was tested. To this effect, the dataset with incomplete gesture length was created. Half length and quarter length incomplete gestures were created by only using the the first sixteen and eight frames from the start of the hand motion. To assure that some hand motion indeed exists, the first two frames of the gesture sequence were always removed.

2.1 Neural Network training

All neural networks used for the experiments were trained on the negative log likelihood cost function and each uses a soft-max projection on the output layer. The networks with single column were trained for three hundred epochs and 2-column network trained simultaneously were trained for five hundred epochs. The number of epochs are chosen according to the convergence performance C3D network on the sixteen frame networks. A Negative log likelihood cost function was used for calculation of loss on each training. In case of the single phase network the training was completed in three hundred epochs. Owing to their larger size the 2-column networks are trained for five hundred epochs. One epoch is defined as the number of batches required for iteration over the full training set.

3 THE MULTI-COLUMN MODELS

3.1 The components of the Multi-column networks

The C3D layer uses volumetric convolutions. A C3D network learns 3D-filters, the two dimensions

 ISBN 978-80-86943-41-1

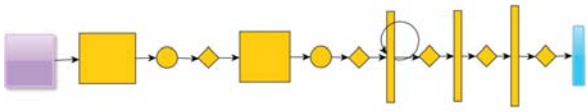

Figure 4: Single column, two input channel LSTM.

of these filters are along the dimension of the frame image and the third dimension is along time axis of the video. Accordingly, the input to the volumetric convolution layer is a block of video frames. As the convoluted blocks are propagated forward through the max-pooling layers the learnt filters reside on higher scales space. Effectively, the minimum time frame of learning in C3D is thus the time length of the spatio-temporal filter on the layers closest to the input. The LSTM on the other hand accepts sequential input. The LSTM learns to use forget gates and identifies the length of the learnt structure in the training phase.

3.1.1 C3D Network

The 2-column network of [MGKK15] uses two networks with high and low resolution input. These networks provide two sets of predictions, which are multiplied, normalized and used for a combined prediction. The C3D network used in this work is shown in Figure 3. The C3D consists of four volumetric-convolution layers, each of these layers have associated volumetric pooling layers. The tanh layers are used as the activation functions after the volumetric convolution. The fourth volumetric convolution feeds into fully connected layers which feed the outputs to the softmax layer. The softmax layer provides a probability vector as the output. The C3D provides one output for the entire block of the K stitched inputs, the output prediction in case is the index with highest probability.
For designing a network that learns to classify a gesture of length K, the input to the C3D is a $K \times 2 \times 57 \times 125$. The experiments were conducted such that each frame of the input block belonged to the same gesture type. An output probability vector of fifteen gestures is produced at the output of the C3D.

3.1.2 The LSTM Network

The LSTM network in the second column of the network has two convolutional layers followed with the usual pooling and ReLu layers. An LSTM layer and a fully connected layer follows the convolutional layers, see Figure 4. The same $K \times 2 \times 57 \times 125$ input for the C3D is feed into the LSTM. The output layer is a soft-max projection. Each frame of the gesture sequence is marked by a label such that the LSTM produces a probability output at every frame of the gesture. The LSTM predictions is made by cumulative

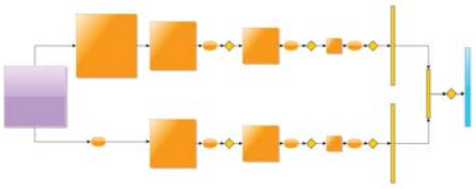

Figure 5: 2-column C3D joined at input and output.

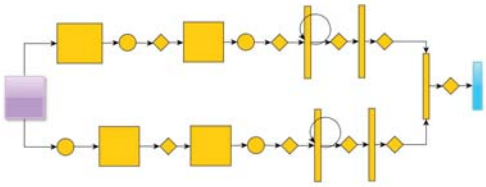

Figure 6: 2-column LSTM joined at input and output.

probability addition over the gesture sequence. The index with highest probability sum at the end of the sequence is identified as the gesture.

3.2 Experiments with Neural Network combinations

We now train , 2-column neural networks, One column of the both the C3D and LSTM networks are as described earlier in . In both cases an average pooling layer is used at the input of the second column, the remaining architecture of the second column remains similar to their corresponding first column. This is done to provide varying scales as input to the first convolutional layers of the two columns of each network. The first volumetric pooling layer in the C3D network scales only in the spatial dimensions and does not change the size of input on the time dimension.

The neural networks are trained with data from fourteen recordings from seven persons and tested on two sets of recording from the eighth person. The final accuracy results are averaged over eight experiments where all the test persons are used once. All networks are trained for full sequence(thirty-two frames) and half and quarter sequences(sixteen and eight frames)

3.2.1 2-Column Neural Networks

We performed end-to-end training with 2-column neural networks based on the components described in the Section 3.1. First, the 2-column C3D was compared against a similar size LSTM network. Thus, the networks trained for these experiments were,

- A 2-column neural network with 3D convolutional layers joined at head with a fully-connected layer, see Figure 5,

ISBN 978-80-86943-41-1

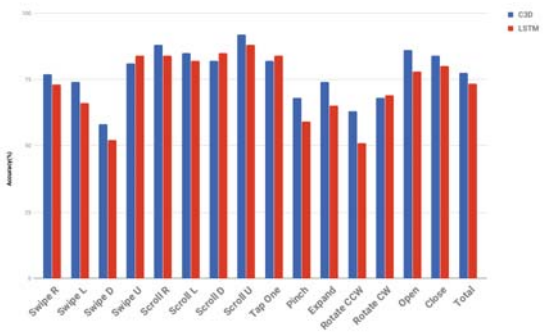

Figure 7: The accuracies of 32 Frame C3D(Red) and LSTM(Blue).

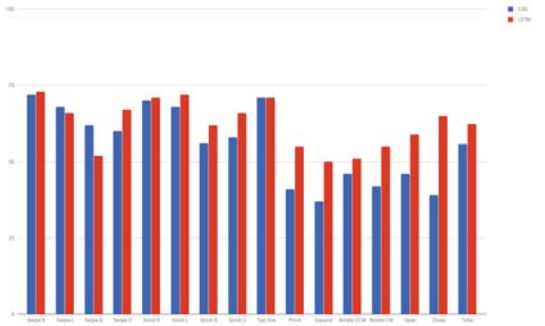

Figure 8: The accuracies of 16 frame sequence on C3D(Red) and LSTM(Blue).

- A 2-column neural network with convolutional layers followed by LSTM layer and joined at head with a fully-connected layer, Figure 6.

These networks were trained and tested for full sequence gestures of length $(K = 32)$ and half and quarter length gestures of frame length $(K = 16, 8)$.

Time Layer	Gesture Frames	Accu (%)
Conv LSTM	32	73.4
	16	**62.3**
	8	**37.3**
C3D	32	**77.4**
	16	55.7
	8	31.6

Table 1: Classification Accuracy with the 2-column LSTM and C3D.

The recorded percentage test accuracies for the two networks for the various frame lengths are reported in the Table 1. The convolutional network with the LSTM layer performed worse than the network with volumetric convolutional layers on the full sequence gestures, though the LSTM network performed better than the C3D network for shorter sequences. The results from these experiments are listed in Table 1. The class-wise classification performance of these networks on the full sequence and half length gestures is shown in the Figure 7 and Figure 8, respectively.

4 TESTING SINGLE COLUMNS

The results from the last section motivated training only the single column C3D networks and LSTM networks to identify if the behavior of volumetric convolutions and LSTM layers remain consistent. These networks were trained with the same set of inputs and labels and the initialization procedures, cost function remained the same as earlier. Apart from the two networks, another network with classical recurrent layer is also tested. The model architectures are exactly like the larger column of the neural network models described in Section 3.1. The three neural networks trained were,

- A neural network architecture from the large column of the convolutional LSTM used in 2-column experiments,
- A similar C3D network taken from the 2-column network gesture classification network,
- A neural network architecture from the large column of the convolutional LSTM used in 2-column experiments with LSTM layer replaced by a recurrent network.

The results of Table 2 demonstrate that the performance of classical recurrent neural network for the classification was poorer compared with the performance of the neural network architectures that use the LSTM layers or the volumetric convolutional layers. This is expected because an RNN network is not capable of learning long contexts.
Looking at the classification performance of Table 2, it is also apparent that the performance of the C3D reduced considerably when an early detection of gesture was made using a C3D network. The performance also deteriorated for networks with convolutional layers and an LSTM layer. An important observation is the considerably smoother decay of performance in the network with LSTM layer as compared to the C3D network. The performances of the LSTM and C3D networks on various datasets is consistent with the observations from the 2-column networks tested earlier. The C3D network performed better on the full length sequence but its performance worsens more rapidly than the LSTM network when tested on incomplete gesture sequences. Thus, both single column and the two column networks results show that the C3D performs better on full sequence gestures and the LSTM network performs better on the shorter sequences.

ISBN 978-80-86943-41-1

Time Layer	Gesture length	Accuracy (%)
Convolutional & Recurrent	32	35.6
	16	18.3
Convolutional & LSTM	32	64.6
	16	51.3
Volumetric conv (C3D)	32	73.6
	16	47.6

Table 2: Accuracy with single phase models

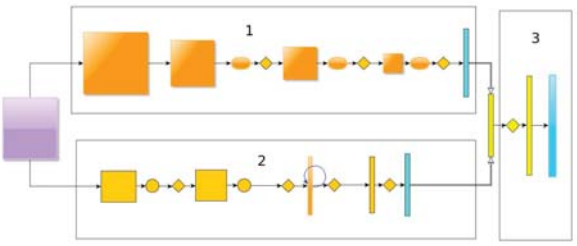

Figure 9: The proposed combination of network : Part 1 is the C3D branch; Part 2 is the LSTM branch; Part 3 is the MLP which combines output from the two temporal neural networks.

5 LSTM AND C3D COMBINATION

The observations that C3D consistently perform better on long sequence gestures, while the LSTM network always works better than C3D on shorter sequences encourages the experiments with combinations of the C3D with LSTM. The trained single phase LSTM and C3D networks were used. The output probabilities of these trained networks were combined with a separately trained MLP. The MLP learns to combine the output of the probability predictions made by the two separate networks.

The cumulative sum of the LSTM was normalized and a larger thirty dimensional vector was created by merging this resulting vector with the C3D output. The MLP is trained with an input of a thirty vector input; the output is the probability vector. The entire system is shown in Figure 9.

5.1 Training the MLP

The fifteen dimensional probability vector from the C3D and LSTM are combined together to form a thirty vector input to the MLP. The MLP has a hidden layer with sixty four nodes and an output layer of fifteen which is mapped to the softmax values. The labels of the C3D are used to train the MLP. The MLP combines the classification probability from the two networks and uses a learning rate of 0.01, is trained for two hundred epochs.

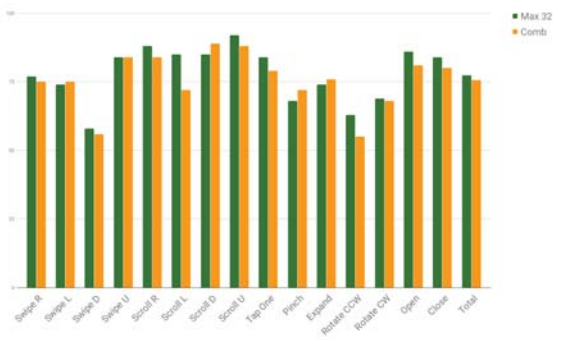

Figure 10: Class-wise average performance of 32 frame hand gestures on the Combination Network (Gold) Compared against the Best of C3D and LSTM Network (Green).

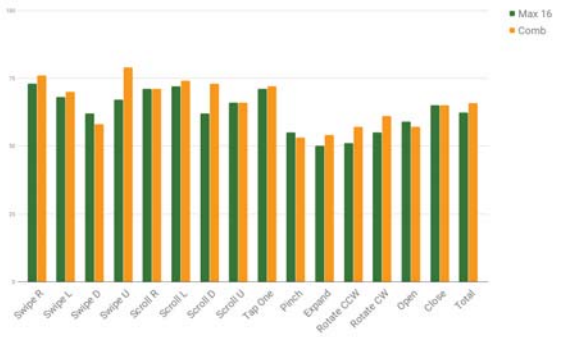

Figure 11: Class-wise average performance of 16 frame hand gestures on the Combination Network(Gold) Compared against the Best of C3D and LSTM Network(Green).

6 PERFORMANCE COMPARISON

To validate the proposed combination network, various training and test iterations were made. The networks were trained with reducing latency time. The MLP was trained separately for full length gesture of thirty-two frames, half length sequence of sixteen and for quarter length of eight frame latency. These results were compared with the best results received from either LSTM or C3D 2-column networks. The class-wise accuracies for the trained MLP network are reported in the Figure 10 and Figure 11. The accuracies are plotted for the thirty-two and sixteen frame gestures respectively. The comparisons show that the performance of the combination network is better than either of the two networks in most classes in the incomplete gesture identification problem. It was observed that the combination network had better accuracy than the best of LSTM and C3D in seven of the fifteen classes when the experiments were conducted on the full length gestures. On the other hand, when the experiments were conducted on the half gesture length starting from

ISBN 978-80-86943-41-1

Gesture length	Combination NN Accuracy(%)	LSTM (%)	C3D (%)
32	75.6	73.4	**77.4**
16	**65.7**	62.3	55.7
8	**39**	37.3	31.6

Table 3: Classification with the combination of C3D and LSTM compared with LSTM and C3D; the accuracy of best network is bold.

the beginning the performance of the combination network was better than the best of the two networks on twelve of the fifteen classes.
The average accuracies achieved in the the experiments conducted on the full, half and quarter gesture lengths are reported in the Table 3. These values are compared against the performances of the 2-column LSTM and C3D tested in the Section 3.

The results of this combinatorial networks tabulated in Table 3 demonstrate that the network performs slightly worse than the two column C3D network in case of long gestures. However, the combination network outperformed the two-column LSTM based gesture classifier in every scenario. When classification accuracies were evaluated at shorter latency period it was observed that the combinational network performed better than the 2-column C3D network. For a half length gesture sequence the accuracy of the combinatorial network was 10% higher than the C3D network (reported in Table 3), it was also marginally better than the LSTM network by 3%.
The combination of the block learning property of the C3D with the contextual learning of the LSTM network may explain the improved performance of the network on shorter incomplete sequences. The accuracy results for the experiments conducted on the one-fourth length sequences demonstrated similar results. The results of the quarter gesture dataset also demonstrated the difficulty of early identification of the gestures. It is clear that the accuracy rates falls dramatically as the sequence length is reduced.

7 DISCUSSION AND CONCLUSION

When the models are tested in the forward phase on a CPU the proposed network with sequential input to the MLP does not return a real time performance. A smaller model means less computation cost in a system embedded in the automobile. More importantly, we wished to understand the generalization behavior on reducing the size of a 2-column neural network for gesture recognition. So, apart from the large combinational model, we trained a smaller model on the same dataset. It included two volumetric convolution layers and two linear layers apart from the output log softmax layer in one branch, and one volumetric-convolutional layer, and a fully connected layer in the other branch. It was identified that choice of the initial learning parameters for a smaller network is crucial. The performance of such a network was generally worse.
We tested this network for 32 frame and 16 frame gesture classification problem. It was recognized that that a combinational network with smaller contributing networks performs considerably worse than the larger network.

Gesture length	Accuracy(%)
32	53
16	42

Table 4: Classification with the combination of smaller C3D and LSTM Networks.

7.1 Conclusion

This work showed a possible method for improving fast identification of hand-gestures. It proposed a possible combination of the C3D and an LSTM network. We show an improvement in the early classification performance. The proposed combinational network performs better as compared to existing state-of-art C3D neural networks by over 10% when applied for early identification of hand gesture sequences. It is shown that the C3D network performs better than LSTM on fixed length full gesture sequence, but LSTM performs better than the C3D network on incomplete sequences.
We demonstrated that a combination of such sequential learning and time filtering networks can improve the classification performance on shorter sequences.
The model for the combination of C3D and LSTM can be extended further and the proposed example should encourage further investigations. This work uses discontinuous windows while training and testing the model. This choice is constraint to a fixed input size. It is possible to use a sliding window approach for sampling while training and testing. Such an approach would allow working with gestures of variable sizes. A system of this nature should have the capacity to handle unsegmented gestures.

8 ACKNOWLEDGMENT

This work is supported by the National Research Fund, Luxembourg, under the AFR project 7019190.

9 REFERENCES

[AAGES10] Nesreen K Ahmed, Amir F Atiya, Neamat El Gayar, and Hisham El-Shishiny.

 ISBN 978-80-86943-41-1

An empirical comparison of machine learning models for time series forecasting. *Econometric Reviews*, 29(5-6):594–621, 2010.

[CFH03] Feng-Sheng Chen, Chih-Ming Fu, and Chung-Lin Huang. Hand gesture recognition using a real-time tracking method and hidden markov models. *Image and vision computing*, 21(8):745–758, 2003.

[CLS15] Guilhem Chéron, Ivan Laptev, and Cordelia Schmid. P-cnn: Pose-based cnn features for action recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3218–3226, 2015.

[CMS12] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. Multi-column deep neural networks for image classification. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 3642–3649. IEEE, 2012.

[DAHG+15] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2634, 2015.

[GYB04] S Burak Gokturk, Hakan Yalcin, and Cyrus Bamji. A time-of-flight depth sensor-system description, issues and solutions. In *Computer Vision and Pattern Recognition Workshop, 2004. CVPRW'04. Conference on*, pages 35–35. IEEE, 2004.

[KKKA13] Cem Keskin, Furkan Kıraç, Yunus Emre Kara, and Lale Akarun. Real time hand pose estimation using depth sensors. In *Consumer Depth Cameras for Computer Vision*, pages 119–137. Springer, 2013.

[LGS08] Yun Liu, Zhijie Gan, and Yu Sun. Static hand gesture recognition and its application based on support vector machines. In *Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, 2008. SNPD '08. Ninth ACIS International Conference on*, pages 517–521, Aug 2008.

[MGKK15] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Jan Kautz. Hand gesture recognition with 3d convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1–7, 2015.

[MGKP15] Pavlo Molchanov, Shalini Gupta, Kihwan Kim, and Kari Pulli. Multi-sensor system for driver's hand-gesture recognition. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 1, pages 1–8. IEEE, 2015.

[OBT14] Eshed Ohn-Bar and Mohan Manubhai Trivedi. Hand gesture recognition in real time for automotive interfaces: A multimodal vision-based approach and evaluations. *IEEE Transactions on Intelligent Transportation Systems*, 15(6):2368–2377, 2014.

[RSP11] Yvonne Rogers, Helen Sharp, and Jenny Preece. *Interaction design: beyond human-computer interaction*. John Wiley & Sons, 2011.

[TTGS16] Aditya Tewari, Bertram Taetz, Frederic Grandidier, and Didier Stricker. Two phase classification for early hand gesture recognition in 3d top view data. Springer, 2016.

[viv] Viva. http://cvrr.ucsd.edu/vivachallenge/index.php/hands/hand-gestures/.

[WKSL13] Heng Wang, Alexander Kläser, Cordelia Schmid, and Cheng-Lin Liu. Dense trajectories and motion boundary descriptors for action recognition. *International journal of computer vision*, 103(1):60–79, 2013.

[YH15] Jiachen Yang and Ryota Horie. An improved computer interface comprising a recurrent neural network and a natural user interface. *Procedia Computer Science*, 60:1386–1395, 2015.

 ISBN 978-80-86943-41-1