A Probabilistic Combination of CNN and RNN Estimates for Hand Gesture Based Interaction in Car

Aditya Tewari*
TU Kaiserslautern, Germany.
IEE SA, Luxembourg.

Bertram Taetz†
TU Kaiserslautern, Germany.

Frederic Grandidier‡
IEE SA, Luxembourg.

Didier Stricker§
German Research Center for Artificial Intelligence, Kaiserslautern, Germany.

ABSTRACT

Hand Gesture Recognition is completed on top-view hand images observed by a Time of Flight(ToF) camera in a car. The work attempts to solve two important problems of touchless interactions inside a car. First, low latency identification of the gestures which are unobtrusive for the driver. Second, reducing the labelled data required to train learning based solutions, this is particularly important because labelling of gesture sequences is expensive and exigent.

This work attempts to improve the fast detection of hand-gestures by correcting the probability estimate of a Long Short Term Memory (LSTM) network by pose prediction made by a Convolutional Neural Network(CNN). Weak models for hand gesture classes based on five hand poses are designed to assist in the prediction-correction scheme. A training procedure to reduce the labelled data required for hand pose classification is also introduced. This method tries to utilise the statistical property of the dataset to identify a good initialization of weights for the CNN, here we demonstrate this using the Principal Component Analysis(PCA) embedding of non-labelled hand pose sequences. While solving a nine class hand gesture problem we demonstrate an accuracy of 89.50% which is better than existing systems. We also show that a PCA embedding based initialization improves the classification performance of the CNN based pose classifier.

Index Terms: I.5.4 [Computer Vision]: ;— [H.5.2]: Interaction styles—; H.m [Hand Gestures]: Hand Pose—; I.4.m [Activity recognition and understanding]: ;— [I.5.1]: Neural nets—LSTM,CNN

1 INTRODUCTION

Ease of interaction with the elements of augmented reality in the real space is of much research interest. Both, posture and gesture of body and hand are simple interaction tools for an augmented reality application. The benefits of Head-Up Displays(HUD) and contactless dashboards inside an automobile have been discussed for decades [17], [8]. Researchers have aimed at reducing the distraction of interaction and a minimum eye contact time for improving driver safety. With increasing research on HUD [15] and the arrival of proposed augmented reality display systems [9] it is important to develop convenient hand gesture systems to interact with both augmented reality and classical applications inside a car. The work by [19] shows that both on-wheel gestures and head-up displays can be integrated. This work shows that frequently used controls in a car can be displayed on a head up display and can be controlled by extending a specified number of fingers without leaving the steering wheel.

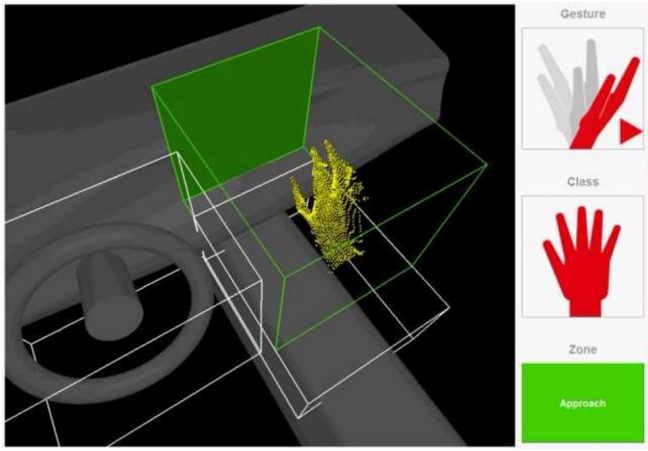


Figure 1: Grab of application for hand pose and gesture in car environment.

Work by [13] also shows that the performance of the driver can degrade sharply with small increase in the shift of attention, thus touch- free gesture interaction based on vision can improve the safety of the driver. It has been shown in these studies that the in vehicular interactions are most safe when they are close to the natural gestures and are simple [16].

More flexibility is allowed while designing a hand-gesture vocabulary which uses varying hand pose and hand motion over frames. We will define such gestures as dynamic gestures. There is a broad range of solutions for interaction using such hand-gestures. Early solutions of dynamic hand gestures include solutions based on finite state machines [4]. Inspired by the results on handwriting recognition [14] and speech analysis [25], various adaptations of Hidden Markov Models(HMM) for gesture recognition systems have been used [3]. Such works used a feature space to learn the dynamics of each action class [21]. Recurrent neural networks have also been used for hand gesture classification extensively [28]. Both the state machine and Recurrent Neural Network(RNN) based strategies use the information of the pose to identify the gestures.

Further, many interaction systems use motionless hand-posture for input interaction. These posture based methods are defined as static hand gestures. In these gestures, the shape of the hand is not expected to modify until the interaction is completed. These methods can use a hand pose classification solution like [5], [20]. Further, posture estimation is another solution for static input interaction with hand. Various works in recent past have used convolutional neural networks(CNN) for hand pose estimation[24].

More recently, deep neural networks have been trained with a multimodal training strategy. Considerable work on language and speech research uses a combination of texts and vision input for the speech recognition and caption generation[10], [18]. It was shown in these works that learning patterns in one feature set is supported by the

*e-mail:aditya[dot]tewari[at]dfki[dot]de
†e-mail:taetz[at]cs[dot]uni-kl[dot]de
‡e-mail:Frederic[dot]Grandidier[at]iee[dot]lu
§e-mail:didier[dot]stricker[at]dfki[dot]de

presence of another feature set. For pose and gesture estimation these multimodal datasets may include a bag of skeletal points, inertial measurements from on-body sensors or a sequence of image frames. Such networks use convolution based networks for image input and the LSTM version of the RNN for inertial inputs.
The LSTM network [12] is a variation to the traditional RNNs. It attempts to solve the problem of vanishing gradients [11] by adding a constant error carousel and forget gate to the recurrent nodes in the neural network and has shown exceptional results on various applications[6],[7]. It is also easier to construct an LSTM as compared to an HMM where some prior experiments are required to identify the number of states. The LSTM has been shown to outperform the traditional RNN and has been extensively used for handwriting and speech recognition tasks recently [6], [7]. In [23] it has been shown that LSTM performs better than HMM and SVM for gesture identification. Location, orientation and velocity of the palm have been used as features for gesture recognition problem in [29]. The union of these features have been used as input to RNN or HMM.
The performance of the image and vector input multi model system has been very encouraging. However, the end to end learning of multimodal networks remains a hard task. The large datasets are required for training complex models with gesture sequences because the hand gestures are complex, the high degree of freedom motion. It is hard to label and train gesture sequences of varying lengths, this is because the segmentation of the sequences is time-consuming with labelling the sequences. Some of the common solutions for training with unlabeled data in an unsupervised manner or by simplifying the learning model being used.

2 Contributions and Structure

2.1 Contributions

Rather than learning an end-to-end multimodal system because of the above-mentioned constraints, we propose to combine separately trained networks. An LSTM based on a vector with motion and shape features and a CNN learnt on the image input. The output class probabilities are used in a prediction correction scheme using a state probability model learnt statistically. The hand gestures can be defined as series of hand motions with changing or static hand poses. In this work, it is argued that the nine gestures we classify can be weakly modelled by transition probabilities of five hand poses. Using this assumption for the definition of gestures we describe a method to correct the gesture estimate provided by the trained LSTM model simultaneously with pose estimate from the CNN model.
Thus we use a smaller model that can be trained with lesser labelled data. We train an LSTM for gesture classification on features extracted from frame sequences. This model was earlier used to make early predictions of hand gestures in [27]. The contributions of this work are:

- A CNN model is used for class probability estimation. We use a novel PCA projection based initialisation for the CNN. This allows us to use the larger dataset of unlabeled hand shapes collected from gesture sequence recordings to initialise the CNN correctly. The initialised CNN is then fine tuned with the labelled hand pose frames.

- Transition probability models describing the pose transition for each gesture are constructed.

- The LSTM prediction is corrected by the CNN pose probability estimate using the pose transition models. We demonstrate that the LSTM predictions for short gesture sequences can be improved by including pose estimation while making a final gesture estimate.

Figure 2: The camera (shown in red box) in the car.

The LSTM architecture is based on a combination of two phase learning and a cumulative probability addition at the output. The work demonstrates improved gesture classification results. The pose accuracy is also improved by using the proposed initialisation scheme.

2.2 Structure of the Paper

The Section 3 introduces the gestures used for the experiments and describes the data collection process. The next section describes two separate training data for the CNN and LSTM and the test data for the entire system. The LSTM and the CNN architectures are described in Section 4. In the same section, the PCA based initialisation of the CNN pose classifier is discussed. The Section 5 explains the correction of gesture probability estimates from the LSTM using the pose probabilities and the probability models. The results of the gesture recognition system on a test data are reported in Section 6, here we compare the results with the original LSTM and demonstrate an improvement in performance. A brief discussion about convolution based recurrent networks and 3D-Convolutional Neural Network(3DCNN) is presented in the Section 7. Finally, we conclude with the limitations and the direction of the work in Section 8.

3 Pose and Gesture Data and Features

Handling the consistently and rapidly changing global illumination and defining an optimal location for the camera to minimise the palm occlusion needs both algorithmic solution and planning on location and choice of the sensor. It has been observed that an overhead location is best suited for such problems [2]. While this location of the camera removes the occlusion due to objects inside the car, the self-occlusion of the hand remains a problem. The illumination variance is handled by using a PMD Nano Time of Flight(ToF) camera based on the Photonic Mixer device(PMD) which records the scene independent of the illumination of the environment. The camera is fixed to the region behind the rear view mirror, Figure 2.
Nine hand gestures are recorded. Of these nine gestures 'Accept', 'Decline', 'Drop', 'Grab', 'Click' are labelled with a single

integer. These gestures include a change of hand poses during their completion. The 'Swipe' in Left and Right, 'Up' and 'Down' have two labels one to identify the 'Horizontal' or 'Vertical' motion and the other to mark the direction of motion.
Five poses 'Fist', 'Flat', 'Join', 'Point', and 'Open' are recorded separately. These poses are chosen because the transition between these poses can define our gesture set. The process of describing the gesture is described in Section 5.
Both the hand-pose and hand-gesture datasets are recorded within a cuboidal space with varying heights. The ToF camera is mounted vertically above the recording region. A PMD Nano sensor has a resolution of 120x165 pixels. The output frame has two channels, the depth channel and the amplitude channel. The data is recorded with a frame rate of 25 frames per second. The palm region is segmented by creating a virtual cuboidal space in the region where we wish to observe the hand-gesture.

3.1 Feature Extraction for Gesture Estimation by LSTM

For gesture recognition, various features that capture information of the hand-motion and hand-shape are extracted from the sequence of frames. Motion descriptor features include velocities of the foremost finger and the hand centroid in all three dimensions. The palm-center coordinates, the azimuth and polar angle of the extended finger, the active pixels of the segmented-palm to indicate shape, the convex hull and concave depth of the palm are calculated at every frame.
Overall, a 17 dimension vector is used to describe the palm shape and motion. This input vector sequence is input to the LSTM. These features are centred and normalised such that the mean of each feature element over the training data is zero and the variance is unity. The start and the end of the frame in the input stream is marked both in the test and train data.

3.2 Training and Test Data

60000 sample of labelled images of five pose classes are recorded. The pose classification model is tested on 12000 test samples.
Gesture sequences from 12 subjects are recorded. Each subject is recorded for multiple sequences. Each of the nine gestures is recorded for two to three minutes. The recordings of 10 people were used to train the LSTM system for gesture-recognition. Recordings from two subjects of the 12 recorded are used for testing and reporting the results for the full system. The LSTM gestures are reported for the same two subjects, which are used for the full system.
The gesture sequences are normalised to the length of 30 frames. Shorter and longer gesture sequences are upsampled and downsampled respectively to create equal length sequences for the experiments.

4 CNN and LSTM Classifiers

4.1 A Two Phase LSTM System

We employ the two phase system described in [27]. It uses two LSTM networks such that the classification task is distributed into a classification based on direction and a classification based on hand shape during the completion of the gesture. One LSTM identifies the primary class of the gesture. Two other, smaller LSTMs are trained to learn the intended direction of motion for the gestures whose identificaiton requires the understanding of the direction of motion Figure 3.
These three LSTM networks are trained independently with the same training dataset by sampling with respective labels. These networks are trained with the resilient backpropagation algorithm [26].
The LSTM combination based system provides an estimate of the

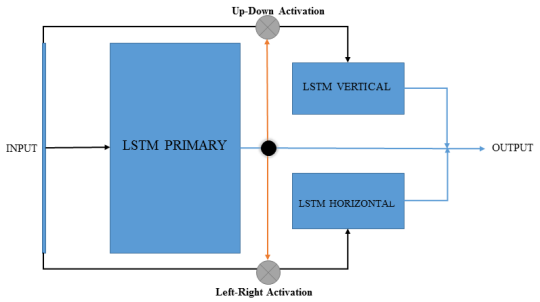


Figure 3: Two phase LSTM architecture.

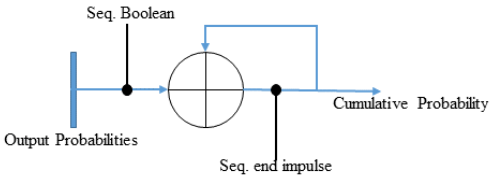


Figure 4: The output strategies.

probability that the frame belongs to a gesture sequence. The output probability values are a cumulative addition of the probabilities from the beginning of the gesture sequence, see Figure 4. The cumulative addition scheme allows continuous estimation of the probability vector. This system provides a continuous gesture probability vector for the ongoing sequence. The LSTM probability addition smoothens the LSTM estimates and suppresses the output during the period of short context.

4.2 CNN Pose Classifier

After experimenting with various three convolutional layer CNN models for hand-pose classification the sequential CNN shown in Figure 5 is identified as the best performing network. This network does pose classification by forcing the outermost layer before output softmax layer to be a 22 node layer which equals the degree of freedom of the hand. The selected architecture has four convolution layers followed by four fully connected layers which calculate the inner product. To add non-linearity to the network each convolutional and fully connected layer before the layer is connected to a ReLu (Rectified linear unit).

The neural network is trained to optimise the performance with the limited labelled hand pose frames. The network with the convolutional layers is trained to learn a twenty-two dimension low dimensional representation of the hand pose images. The frames from hand gesture sequences are used to learn a PCA low dimensional representation. Ten-thousand frames are used to learn a PCA decomposition. Further, the network is trained as a regressor with sixty-five thousand image frames. The low dimension representation is calculated by the learnt PCA embedding. These projections are the labels to be learnt by the algorithm. The PCA-embedding network is trained using a batch learning scheme with a batch size of 64. The PCA embedding is tested on six thousand frames from the same gesture dataset.
Finally, a five node softmax layer is added to the PCA-embedding network and is fine tuned to learn a classifier. The model initialized by this procedure learns the classification task faster. The accuracy gain is demonstrated in the section on results, while the loss progression in PCA initialised and the original model is shown in Figure 8 and Figure 7, respectively. Lesser iteration over the batches

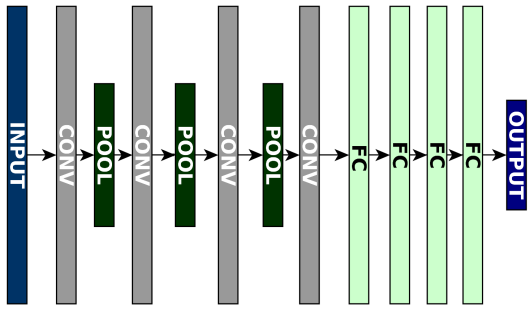


Figure 5: The neural network model for pose recognition.
Conv:convolutional, FC:fully connected.

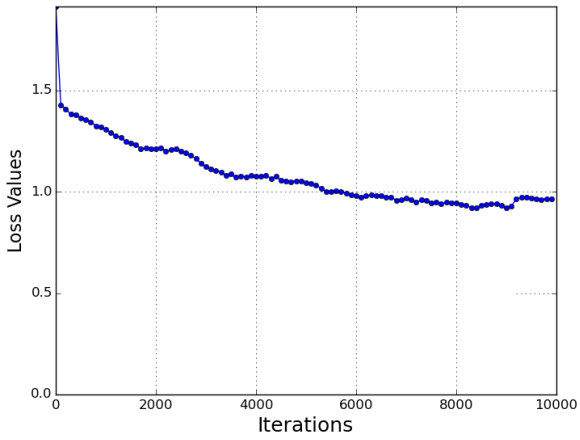


Figure 6: Test loss progression on PCA embedding network.

of data produces better loss performance in case of the initialized network.

5 Correction Estimation

The gesture estimates for the LSTM are made for every frame. Similarly, the pose class estimates are available for every frame of the gesture. Let ρ_G and ρ_P be the respective output probability vectors for the gesture=classes and pose-classes.

5.1 Gesture models

A small dataset with twenty gestures from each class is created. The dataset has a gesture label for the full sequence. Apart from the gesture label, each frame of the normalised gesture sequence is labelled with the closest hand pose. This dataset is used to create a probability model $\rho_{P|(G=g)_t}$, to be read as probability of a pose occurring during the t^{th} part of the sequence given gesture g.
Each gesture is modelled as a sequence of five pose probabilities over a time frames of six frame each.
The pose appearing over each time frame in the samples of each dataset are counted and a probability estimate is made. To avoid zero-probability situations a value ε is added to each probability value in the models. The probabilities are renormalised after this addition. The sample probability calculated model of gestures 'accept' and 'decline' are shown in Figure 10a and Figure 10b.

Figure 7: Classification model initialised from PCA

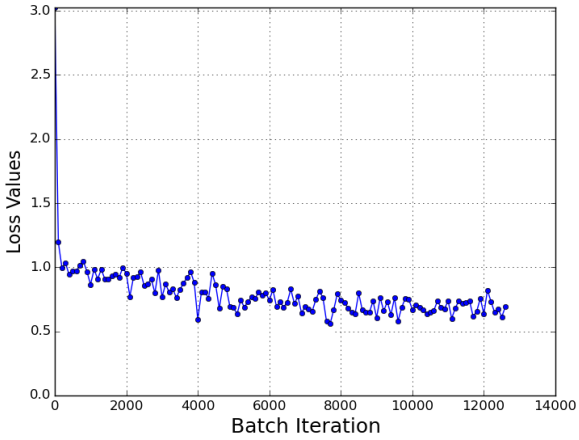


Figure 8: Classification without initialization

5.2 Correction

We have a probability vector estimate of the gestures from the LSTM output (ρ_G). Another estimate for the gesture probability vector $\rho_G^{'}$ is made using the conditional probability $\rho_{P|(G=g)}$ and the pose probability output ρ_P of the CNN.
Further, we construct a $\rho_{G|(P=p)_t}$ and use the gesture estimate from the LSTM ρ_P to make a new pose estimate $\rho^{'}{}_P$.
A deviation measure for $\rho_P^{'}$ and ρ_P is defined to measure the divergence of the pose estimate by the gesture model from the trained CNN. The pose probability divergence is used for the online re-weighting of the iterative weighting factor. The weighted gesture probabilities at each frame are then calculated as,

$$\rho(G_{comb})_f = \rho_G + (\psi(\delta(\rho_P, \rho_P^{'}))\rho_G^{'}, \tag{1}$$

where $\psi(x) = 1 - x$ and,

$$\delta = max(\rho_P) - \rho_P^{'}[arg_{max}\rho_P]. \tag{2}$$

This is done to give high weighting to the LSTM estimates if the strong classifier, based on CNN, is more certain about a class as

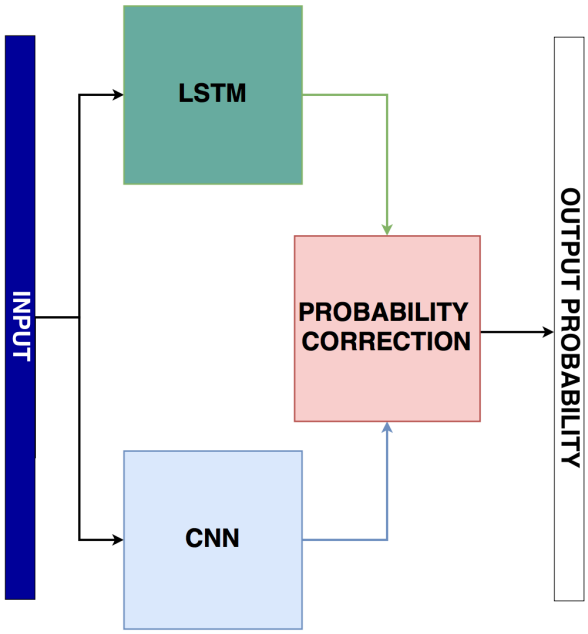
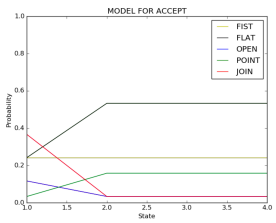
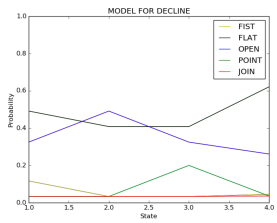


Figure 9: The combined system for gesture sstimation.



(a) Pose probability model for "Accept"



(b) Pose probability model for "Decline"

compared to the weak model. This corrects for unintended, short change of pose while completing a gesture. The weighted gesture probability is renormalised.

6 Results

The LSTM model for gesture recognition and the CNN model for the pose recognition were trained and tested separately. Further, the probability output of these models was combined to create a better estimate of the sequence. The results on both training modes for CNN classifier from two separate gesture and pose dataset are listed. The corrected estimation for the gesture after combining the probabilities from the two models are also reported in this section.
The results obtained from our method were compared against the 2-phase LSTM system that was used for making a gesture estimate. The gesture classification accuracies were also compared with a single all class LSTM which was trained and tested on the same dataset.
The accuracy performance improved when the CNN for pose classification is trained with the proposed initialization based on first training the network with the PCA embedding. The initialized network was trained for 5200 batch iterations as compared to the Xavier initialised network which was trained for 12700 batch iterations. The modified training procedure improves the classification accuracy for all classes of hand poses. This demonstrated a suitable solution for better classification when the labelled data is scarce or expensive. The new training procedure improved the average classification accuracy of the full test dataset from 90.5% to 91.5%. The class-wise accuracy for both initializations are shown in Table 1.

Table 1: Accuracy for classification of five poses with two initialisations.

Accu %	Fist	Flat	Open	Point	Join
Xavier Init	87	89	91	91	93
PCA Init	**89**	**90**	**94**	**91**	**93**

Table 2: Accuracy for nine gesture classification based on three methods.
U:Up, D:Down, L:Left, R:Right, C:Click, A:Accept, De: Decline, G:Grab, Dr:Drop.

% Accu	U	D	L	R	C	A	De	G	Dr
	Static gestures					Dynamic Getsures			
LSTM	77	78	88	89	96	78	80	91	91
2 Phase LSTM	**84**	85	**92**	93	**96**	82	84	89	91
Proposed	81	**86**	89	**93**	95	**85**	**87**	**92**	**92**

The two-phase LSTM described earlier, is trained for 200 epochs, the classification decisions are made only after the first 10 frames (one third of the normalised gesture length) of the gesture. The accuracy percentages are reported as percentage of correct predictions after the first 10 frames. It is observed that the short or rapid change of gestures were missed because of the LSTM learning procedure. This problem was solved with the correction scheme. The results from a single phase LSTM were compared with the LSTM combination explained earlier and finally with the probability correction paradigm that has been proposed in this work.
The proposed solution outperformed a large single LSTM consistently. On comparison with the 2-phase LSTM the solutions provided better accuracy in six of the nine classes, see Table 2. The proposed solutions consistently performed better on the dynamic gestures. The overall accuracy on the test dataset increased from 88.50% on the 2 phase system to 89.50% on the proposed solution.

7 Discussion

The experiments on the same dataset were also conducted by the combination of image inputs into two parallel networks on 3DCNN and LSTM models like those proposed in [22]. The networks perform well on RGBD datsets [1] collected from Kinect. We could not reproduce similar results when testing on the relatively smaller amount of labelled data recorded from the ToF camera. In the test that we conducted with 4500 sequences, the average classification performance on the nine class problem was 64%. This can both be attributed to the low resolution of the input images and the smaller size of the dataset.

8 Conclusion

We tried to improve the solution for HCI inside car with focus on low latency and low data requirement. A gesture recogntion system that returns gesture estimation on the completion of one-third of the geture is tested and presented. It is expected that a low latency solution for gesture identification is more compatible for use in a car, especially because it is less obtrusive for the driver to use.
This work classified top-view hand gestures observed by a ToF camera using a probabilistic combination of gesture and pose estimates. The work demonstrated the possible starategies for obtaining better classification accuracies with smaller labelled and

unlabelled datasets. We could show an average 1% improvement of performance over multi-phase LSTM based system and over 5% improvement over one LSTM based classifier. The performance improvement is significantly higher for gesture sequences in which the hand pose modifies during completion. The identification accuracy for "Accept", "Decline", "Grab" and "Drop" increase substantially. The dynamic gestures as defined earlier, thus perform considerably better in the new proposed setup.

Further, multiple hand frames in each gesture are unlabelled for pose. It has been demonstrated that learning a PCA embedding of these hand frames through a neural network helps in a better initialisation of the network. It has been demonstrated that the network trained on PCA embedding based initialisation improves the accuracy for the pose classification problem when trained for half the original iterations.

One of the constraints of the solution is the normalisation of the sequence length. This is done to design a constant length state transition model. Additions to this work will work towards handling this constraint by attempting to learn the transition models using belief networks. The PCA embedding based learning also enforces a linear mapping between the input and output. It is planned to experiment further with projections like with local linear embedding and isomap projections.

ACKNOWLEDGEMENTS

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

REFERENCES

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

[2] M. Alpern and K. Minardo. Developing a car gesture interface for use as a secondary task. In *CHI '03 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '03, pages 932–933, New York, NY, USA, 2003. ACM.

[3] F.-S. Chen, C.-M. Fu, and C.-L. Huang. Hand gesture recognition using a real-time tracking method and hidden markov models. *Image and vision computing*, 21(8):745–758, 2003.

[4] J. Davis and M. Shah. Recognizing hand gestures. In *Computer VisionECCV'94*, pages 331–340. Springer, 1994.

[5] W. T. Freeman and M. Roth. Orientation histograms for hand gesture recognition. In *International workshop on automatic face and gesture recognition*, volume 12, pages 296–301, 1995.

[6] A. Graves, A.-r. Mohamed, and G. Hinton. Speech recognition with deep recurrent neural networks. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 6645–6649. IEEE, 2013.

[7] A. Graves and J. Schmidhuber. Offline handwriting recognition with multidimensional recurrent neural networks. In *Advances in Neural Information Processing Systems*, pages 545–552, 2009.

[8] P. Green. Visual and task demands of driver information systems. Technical report, 1999.

[9] S. D. Green, M. R. N. SMITH, J. Pavitt, et al. Display control system for an augmented reality display system, Nov. 25 2015. US Patent App. 14/951,540.

[10] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates, et al. Deep speech: Scaling up end-to-end speech recognition. *arXiv preprint arXiv:1412.5567*, 2014.

[11] S. Hochreiter. The vanishing gradient problem during learning recurrent neural nets and problem solutions. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 6(02):107–116, 1998.

[12] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[13] W. J. Horrey. Assessing the effects of in-vehicle tasks on driving performance. *Ergonomics in Design: The Quarterly of Human Factors Applications*, 19(4):4–7, 2011.

[14] J. Hu, M. K. Brown, and W. Turin. Hmm based online handwriting recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 18(10):1039–1045, 1996.

[15] A. Ishizaki, S. Ikegami, T. Yamabe, S. Kitagami, and R. Kiyohara. Accelerometer-based hud input for car navigation. In *2014 IEEE International Conference on Consumer Electronics (ICCE)*, pages 278–279, Jan 2014.

[16] M. G. Jæger, M. B. Skov, N. G. Thomassen, et al. You can touch, but you can't look: interacting with in-vehicle systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1139–1148. ACM, 2008.

[17] N. Kaptein. Benefits of in-car head-up displays. 1994.

[18] R. Kiros, R. Salakhutdinov, and R. S. Zemel. Unifying visual-semantic embeddings with multimodal neural language models. *arXiv preprint arXiv:1411.2539*, 2014.

[19] S. H. Lee, S.-O. Yoon, and J. H. Shin. On-wheel finger gesture control for in-vehicle systems on central consoles. In *Adjunct Proceedings of the 7th International Conference on Automotive User Interfaces and Interactive Vehicular Applications*, AutomotiveUI '15, pages 94–99, New York, NY, USA, 2015. ACM.

[20] Y. Liu, Z. Gan, and Y. 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.

[21] F. Lv and R. Nevatia. *Recognition and Segmentation of 3-D Human Action Using HMM and Multi-class AdaBoost*, pages 359–372. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[22] P. Molchanov, S. Gupta, K. Kim, and J. 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.

[23] N. Neverova, C. Wolf, G. Paci, G. Sommavilla, G. W. Taylor, and F. Nebout. A multi-scale approach to gesture detection and recognition. In *Computer Vision Workshops (ICCVW), 2013 IEEE International Conference on*, pages 484–491. IEEE, 2013.

[24] M. Oberweger, P. Wohlhart, and V. Lepetit. Hands deep in deep learning for hand pose estimation. *arXiv preprint arXiv:1502.06807*, 2015.

[25] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[26] M. Riedmiller and H. Braun. A direct adaptive method for faster backpropagation learning: The rprop algorithm. In *Neural Networks, 1993., IEEE International Conference on*, pages 586–591. IEEE, 1993.

[27] A. Tewari, B. Taetz, F. Grandidier, and D. Stricker. Two phase classification for early hand gesture recognition in 3d top view data. In *International Symposium on Visual Computing*, pages 353–363. Springer, 2016.

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

[29] H.-S. Yoon, J. Soh, Y. J. Bae, and H. S. Yang. Hand gesture recognition using combined features of location, angle and velocity. *Pattern recognition*, 34(7):1491–1501, 2001.