HOW DEEP NEURAL NETWORKS CAN IMPROVE EMOTION RECOGNITION ON VIDEO DATA

Pooya Khorrami[1]*, Tom Le Paine*[1]*, Kevin Brady*[2]*, Charlie Dagli*[2]*, Thomas S. Huang*[1]

[1] Beckman Institute, University of Illinois at Urbana-Champaign
[2] MIT Lincoln Laboratory
[1] {pkhorra2, paine1, t-huang1}@illinois.edu
[2] {kbrady, dagli}@ll.mit.edu

ABSTRACT

We consider the task of dimensional emotion recognition on video data using deep learning. While several previous methods have shown the benefits of training temporal neural network models such as recurrent neural networks (RNNs) on hand-crafted features, few works have considered combining convolutional neural networks (CNNs) with RNNs. In this work, we present a system that performs emotion recognition on video data using both CNNs and RNNs, and we also analyze how much each neural network component contributes to the system's overall performance. We present our findings on videos from the Audio/Visual+Emotion Challenge (AV+EC2015). In our experiments, we analyze the effects of several hyperparameters on overall performance while also achieving superior performance to the baseline and other competing methods.

***Index Terms*—** Emotion Recognition, Convolutional Neural Networks, Recurrent Neural Networks, Deep Learning, Video Processing

1. INTRODUCTION

For several decades, emotion recognition has remained one of the of the most important problems in the field of human computer interaction. A large portion of the community has focused on *categorical models* which try to group emotions into discrete categories. The most famous categories are the six basic emotions originally proposed by Ekman in [1, 2]: anger, disgust, fear, happiness, sadness, and surprise. These emotions were selected because they were all perceived similarly regardless of culture.

Several datasets have been constructed to evaluate automatic emotion recognition systems such as the extended Cohn-Kanade (CK+) dataset [3] the MMI facial expression database [4] and the Toronto Face Dataset (TFD) [5]. In the last few years, several methods based on hand-crafted and, later, learned features [6, 7, 8, 9] have performed quite well in recognizing the six basic emotions. Unfortunately, these six basic emotions do not cover the full range of emotions that a person can express.

An alternative way to model the space of possible emotions is to use a *dimensional* approach [10] where a person's emotions can be described using a low-dimensional signal (typically 2 or 3 dimensions). The most common dimensions are (i) arousal and (ii) valence. Arousal measures how engaged or apathetic a subject appears while valence measures how positive or negative a subject appears.

Dimensional approaches have two advantages over categorical approaches. The first being that dimensional approaches can describe a larger set of emotions. Specifically, the arousal and valence scores define a two dimensional plane while the six basic emotions are represented as points in said plane. The second advantage is dimensional approaches can output time-continuous labels which allows for more realistic modeling of emotion over time. This could be particularly useful for representing video data.

Given the success of deep neural networks on datasets with categorical labels [11, 9], one can ask the very natural question: is it possible to train a neural network to learn a representation that is useful for dimensional emotion recognition in video data?

In this paper, we will present two different frameworks for training an emotion recognition system using deep neural networks. The first is a single frame convolutional neural network (CNN) and the second is a combination of CNN and a recurrent neural network (RNN) where each input to the RNN is the fully-connected features of a single frame CNN. While many works [12, 13, 14, 15] have trained recurrent neural networks (RNNs) on hand-crafted features, few works [16, 17] have considered the effects of using a CNN as input to the RNN. Our work, developed concurrently with [16, 17], demonstrates the benefits of using a CNN+RNN model and also shows how much the CNN and the RNN individually contribute to the overall performance.

The MIT Lincoln Laboratory part of this collaborative work is sponsored by the Assistant Secretary of Defense for Research & Engineering under Air Force Contract FA8721-05-C-0002. The Tesla K40 GPU used for this research was donated by the NVIDIA Corporation.

978-1-4673-9961-6/16/$31.00 ©2016 IEEE

Thus, the main contributions of this work are as follows:

1. We train both a single-frame CNN and a CNN+RNN model and analyze their effectiveness on the dimensional emotion recognition task. We also conduct extensive analysis on the various hyperparameters of the CNN+RNN model to support our chosen architecture.

2. We evaluate our models on the AV+EC 2015 dataset [13] and demonstrate that both of our techniques can achieve comparable or superior performance to the baseline model and other competing methods.

2. DATASET

The AV+EC 2015 [13] corpus uses data from the RECOLA dataset [18], a multimodal corpus designed to monitor subjects as they worked in pairs remotely to complete a collaborative task. The type of modalities include: audio, video, electro-cardiogram (ECG) and electro-dermal activity (EDA). These signals were recorded for 27 French-speaking subjects. The dataset contains two types of dimensional labels (arousal and valence) which were annotated by 6 people. Each dimensional label ranges from $[-1, 1]$. The dataset is partitioned into three sets: train, development, and test, each containing 9 subjects.

In our experiments, we focus on predicting the valence score using just the video modality. Also, since the test set labels were not readily available, we evaluate all of our experiments on the development set. We evaluate our techniques by computing three metrics: (i) Root Mean Square Error (RMSE) (ii) Pearson Correlation Coefficient (CC) and (iii) Concordance Correlation Coefficient (CCC). The Concordance Correlation Coefficient tries to measure the agreement between two variables using the following expression:

$$\rho_c = \frac{2\rho\sigma_x\sigma_y}{\sigma_x^2 + \sigma_y^2 + (\mu_x - \mu_y)^2} \tag{1}$$

where ρ is the Pearson correlation coefficient, σ_x^2 and σ_y^2 are the variance of the predicted and ground truth values respectively and μ_x and μ_y are their means, respectively. The strongest method is selected based on whichever obtains the highest CCC value.

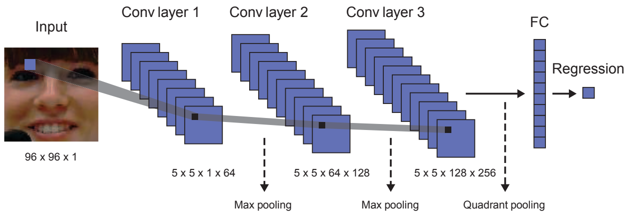

Fig. 1. Single Frame CNN Architecture - Similar to the network in [9], our network consists of three convolutional layers containing 64, 128, and 256 filters, respectively, each of size 5x5 followed by ReLU (Rectified Linear Unit) activation functions. We add 2x2 max pooling layers after the first two convolutional layers and quadrant pooling after the third. The three convolutional layers are followed by a fully-connected layer containing 300 hidden units and a linear regression layer which approximates the valence score.

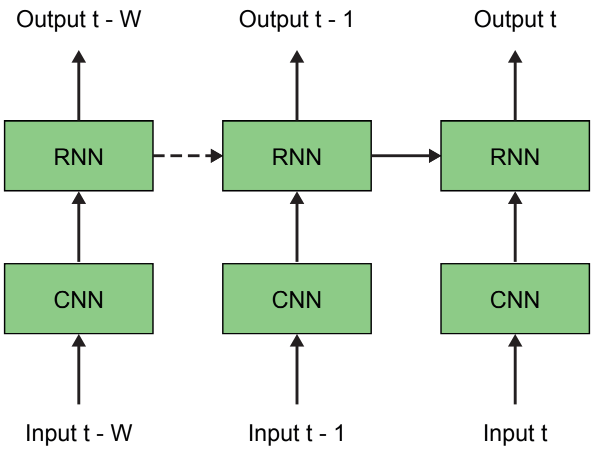

Fig. 2. CNN+RNN Network Architecture - Given a time t in a video, we extract a window of length W frames ($[t - W, t]$). We model our single frame CNN as a feature extractor by fixing all of the parameters and removing the top regression layer. We then pass each frame within the window to the CNN and extract a 300 dimensional feature for every frame, each of which is passed as an input to one node of the RNN. We then take the valence score generated by the RNN at time t.

3. OUR APPROACH

3.1. Single Frame Regression CNN

The first model that we train is a single frame CNN. At each time point in a video, we pass the corresponding video frame through a CNN, shown visually in Figure 1. The CNN has 3 convolutional layers consisting of 64, 128, and 256 filters respectively, each of size 5x5. The first two convolutional layers are followed by 2x2 max pooling while the third layer is followed by quadrant pooling. After the convolutional layers is a fully-connected layer with 300 hidden units and a linear regression layer to estimate the valence label. We use the mean squared error (MSE) as our cost function.

All of the CNNs were trained using stochastic gradient descent with batch size of 128, momentum equal to 0.9, and weight decay of 1e-5. We used a constant learning of 0.01 and did not use any form of annealing. All of our CNN models were trained using the anna software library [1].

[1] https://github.com/ifp-uiuc/anna

3.2. Adding Recurrent Neural Networks (RNNs)

Despite having the ability to learn useful features directly from the video data, the single frame regression CNN completely ignores temporal information. Similar to the model in [17], we propose to incorporate the temporal information by using a recurrent neural network (RNN) to propagate information from one time point to next.

We first model the CNN as a feature extractor by fixing all of its parameters and removing the regression layer. Now, when a frame is passed to the CNN, we extract a 300 dimensional vector from the fully-connected layer. For a given time t, we take W frames from the past (i.e. $[t - W, t]$). We then pass each frame from time $t - W$ to t to the CNN and extract W vectors in total, each length of 300. Each of these vectors is then passed as input to a node of the RNN. Each node in the RNN then regresses the output valence label. We visualize the model in Figure 2. Once again we use the mean squared error (MSE) as our cost function during optimization.

We train our RNN models with stochastic gradient descent with a batch size of 128 and momentum equal to 0.9. All of the RNNs in our experiments were trained using the Lasagne library [2].

4. EXPERIMENTS

4.1. Data Preprocessing

When preparing the video data, we first detect the face in each video frame using face and landmark detector in Dlib-ml [19]. Frames where the face detector missed were dropped. We then map the detected landmark points to pre-defined pixel locations in order to ensure correspondence between frames. After normalizing the eye and nose coordinates, we apply mean subtraction and contrast normalization prior to passing each face image through the CNN.

4.2. Single Frame CNN vs. CNN+RNN

Table 1 shows how well our single frame regression CNN and our CNN+RNN architecture perform at predicting valence scores of subjects in the development set of the AV+EC 2015 dataset. When training our single frame CNN, we consider two forms of regularization: dropout (D) with probability 0.5 and data augmentation (A) in form of translations, flips, and color changes. For our CNN+RNN model, we use a single layer RNN with 100 units in the hidden layer and a temporal window of size 100 frames. We consider two types of nonlinearities: hyperbolic tangent (tanh) and rectified linear unit (ReLU).

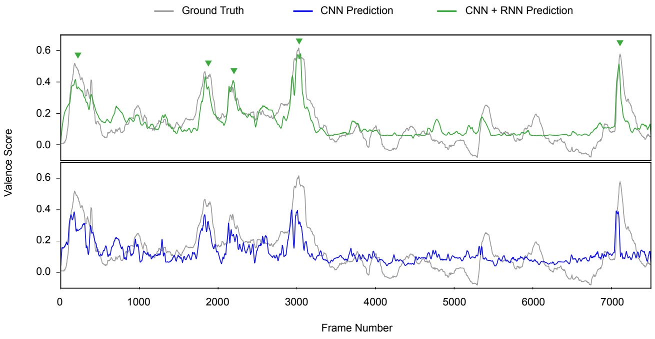

Fig. 3. Valence score predictions of the single frame CNN and the CNN+RNN model for one subject in the AV+EC 2015 development set - Notice that the CNN+RNN model appears to smooth the scores outputted by the single frame CNN and seems to approximate the ground truth more accurately, specifically the peaks (arrows). (Best viewed in color).

Table 1. Performance comparison between: (i) Baseline method with hand-crafted features (ii) Single frame CNN with different levels of regularization (iii) Single frame CNN with an RNN connecting each time point (A = Data Augmentation, D = Dropout)

Method	**RMSE**	**CC**	**CCC**
Baseline [13]	0.117	0.358	0.273
CNN	0.121	0.346	0.248
CNN+D	0.113	0.430	0.331
CNN+A	0.124	0.351	0.272
CNN+AD	0.118	0.408	0.315
CNN+RNN - tanh	0.111	0.523	0.501
CNN+RNN - ReLU	**0.107**	**0.552**	**0.515**

From Table 1, we can see, not surprisingly, that adding regularization improves the performance of the CNN. Most notably, we see that our CNN model with dropout (CNN+D) outperforms the baseline LSTM model trained on LBP-TOP features [13] (CCC = 0.331 vs. 0.273). Finally, when incorporating temporal information using the CNN+RNN model, we can achieve a significant performance gain over the single frame CNN.

In Figure 3, we plot the valence scores predicted by both our single frame CNN and the CNN+RNN model for one of the videos in the development set. From this chart, we can clearly see the advantages of using temporal information. The CNN+RNN model appears to model the ground truth more accurately and generate a smoother prediction than the single frame regression CNN.

4.3. Hyperparameter Analysis

We study the effects of several hyperparameters in the CNN+RNN model, namely the number of hidden units, the length of the temporal window, and the number of hidden

[2] `https://github.com/Lasagne/Lasagne`

Table 2. Effect of Changing Number of Hidden Units

Method	RMSE	CC	CCC
CNN+RNN - h=50	0.110	0.526	0.494
CNN+RNN - h=100	**0.107**	**0.552**	**0.515**
CNN+RNN - h=150	0.111	0.534	0.504
CNN+RNN - h=200	0.108	0.537	0.503

Table 3. Effect of Changing Temporal Window Length (i.e. number of frames used by the RNN)

Method	RMSE	CC	CCC
CNN+RNN - W=25	0.111	0.508	0.485
CNN+RNN - W=50	0.112	0.530	0.501
CNN+RNN - W=75	0.110	0.531	0.504
CNN+RNN - W=100	**0.107**	**0.552**	**0.515**
CNN+RNN - W=150	0.110	0.527	0.496

Table 4. Effect of Changing the Number of Hidden Layers in the RNN

Method	RMSE	CC	CCC
CNN+RNN - W=100 - 1 layer	**0.107**	**0.552**	**0.515**
CNN+RNN - W=100 - 2 layers	0.111	0.526	0.490
CNN+RNN - W=100 - 3 layers	0.107	0.556	0.513

layers in the RNN. The results are shown in Tables 2, 3, and 4 respectively. Based on our results in Table 2, we conclude that it is best to have ≈100 hidden units given that both $h = 50$ and $h = 200$ resulted in decreases in performance. Similarly, for the temporal window length, we see that a window of length 100 frames ($\approx$ 4 seconds) appears to yield the highest CCC score, while reducing the window to 25 frames (1 second) and increasing it to 150 frames (6 seconds) both lead to significant decreases in performance. In Table 4, we see that increasing the number of hidden layers does not improve performance. Thus, based on our experiments, our best performing model had 1 hidden layer with a window length of W=100 frames, 100 hidden units, and a ReLU as its nonlinearity.

4.4. Comparison with Other Techniques

Table 5 shows how our best performing CNN+RNN model compares to other techniques evaluated on the AV+EC 2015 dataset. Both our single frame CNN model with dropout and our CNN+RNN model achieve comparable or superior performance compared to the state-of-the-art techniques. Our single frame CNN model achieves a higher CCC value than the baseline[13] and is comprable with two other techniques [14, 15], all of which use temporal information. While our CNN+RNN model's performance is not quite as strong as the CNN+LSTM model of Chao et al. [16], in terms of CCC value, we would like to point out that the authors used a larger CNN on a larger external dataset. Specifically, the authors trained an AlexNet[20] on 110,000 images from 1032 people in the Celebrity Faces in the Wild (CFW) [21] and FaceScrub datasets [22].

Table 5. Performance Comparison of Our Models versus Other Methods (D: Dropout, W: temporal window length)

Method	RMSE	CC	CCC
Baseline [13]	0.117	0.358	0.273
LGBP-TOP + LSTM [14]	0.114	0.430	0.354
LGBP-TOP+ Deep Bi-Dir. LSTM [15]	0.105	0.501	0.346
LGBP-TOP+LSTM+ϵ-loss [16]	0.121	0.488	0.463
CNN+LSTM+ϵ-loss [16]	**0.116**	**0.561**	**0.538**
Single Frame CNN+D - ours	0.113	0.430	0.331
CNN+RNN - W=100 - 1 layer - ours	0.107	0.552	0.515

5. CONCLUSIONS

In this work, we presented two systems for doing dimensional emotion recognition: a single frame CNN model and a multi-frame CNN+RNN model. We showed that our simple learned representation (single frame CNN) can outperform the baseline temporal model trained on hand-crafted features. With the CNN+RNN model, we showed how incorporating temporal information can yield smoother and more accurate predictions. Lastly, we conducted an extensive hyperparameter analysis and selected a CNN+RNN model that achieved comparable or superior performance to other state-of-the-art emotion recognition techniques on the AV+EC 2015 dataset.

6. REFERENCES

[1] Paul Ekman, Wallace V Friesen, Maureen O'Sullivan, Anthony Chan, Irene Diacoyanni-Tarlatzis, Karl Heider, Rainer Krause, William Ayhan LeCompte, Tom Pitcairn, Pio E Ricci-Bitti, et al., "Universals and cultural differences in the judgments of facial expressions of emotion.," *Journal of personality and social psychology*, vol. 53, no. 4, pp. 712, 1987.

[2] Paul Ekman, "Strong evidence for universals in facial expressions: a reply to russell's mistaken critique.," 1994.

[3] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews, "The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression," in *CVPRW*, 2010, pp. 94–101.

[4] Michel Valstar and Maja Pantic, "Induced disgust, happiness and surprise: an addition to the mmi facial expression database," in *Proc. 3rd Intern. Workshop on EMOTION (satellite of LREC): Corpora for Research on Emotion and Affect*, 2010, p. 65.

[5] Josh M Susskind, Adam K Anderson, and Geoffrey E Hinton, "The toronto face database," *Department of Computer Science, University of Toronto, Toronto, ON, Canada, Tech. Rep*, 2010.

[6] Caifeng Shan, Shaogang Gong, and Peter W McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Image and Vision Computing*, vol. 27, no. 6, pp. 803–816, 2009.

[7] Mengyi Liu, Shaoxin Li, Shiguang Shan, and Xilin Chen, "Au-aware deep networks for facial expression recognition," in *FG*, 2013, pp. 1–6.

[8] Ping Liu, Shizhong Han, Zibo Meng, and Yan Tong, "Facial expression recognition via a boosted deep belief network," in *CVPR*, 2014, pp. 1805–1812.

[9] Pooya Khorrami, Tom Le Paine, and Thomas S. Huang, "Do deep neural networks learn facial action units when doing expression recognition?," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 19–27.

[10] James A Russell and Albert Mehrabian, "Evidence for a three-factor theory of emotions," *Journal of research in Personality*, vol. 11, no. 3, pp. 273–294, 1977.

[11] Samira Ebrahimi Kahou, Christopher Pal, Xavier Bouthillier, Pierre Froumenty, Çaglar Gülçehre, Roland Memisevic, Pascal Vincent, Aaron Courville, Yoshua Bengio, Raul Chandias Ferrari, et al., "Combining modality specific deep neural networks for emotion recognition in video," in *ICMI*, 2013, pp. 543–550.

[12] Fabien Ringeval, Florian Eyben, Eleni Kroupi, Anil Yuce, Jean-Philippe Thiran, Touradj Ebrahimi, Denis Lalanne, and Björn Schuller, "Prediction of asynchronous dimensional emotion ratings from audiovisual and physiological data," *Pattern Recognition Letters*, 2014.

[13] Fabien Ringeval, Björn Schuller, Michel Valstar, Shashank Jaiswal, Erik Marchi, Denis Lalanne, Roddy Cowie, and Maja Pantic, "Av+ ec 2015: The first affect recognition challenge bridging across audio, video, and physiological data," in *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2015, pp. 3–8.

[14] Shizhe Chen and Qin Jin, "Multi-modal dimensional emotion recognition using recurrent neural networks," in *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2015, pp. 49–56.

[15] Lang He, Dongmei Jiang, Le Yang, Ercheng Pei, Peng Wu, and Hichem Sahli, "Multimodal affective dimension prediction using deep bidirectional long short-term memory recurrent neural networks," in *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2015, pp. 73–80.

[16] Linlin Chao, Jianhua Tao, Minghao Yang, Ya Li, and Zhengqi Wen, "Long short term memory recurrent neural network based multimodal dimensional emotion recognition," in *Proceedings of the 5th International Workshop on Audio/Visual Emotion Challenge*. ACM, 2015, pp. 65–72.

[17] Samira Ebrahimi Kahou, Vincent Michalski, Kishore Konda, Roland Memisevic, and Christopher Pal, "Recurrent neural networks for emotion recognition in video," in *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*. ACM, 2015, pp. 467–474.

[18] Fabien Ringeval, Andreas Sonderegger, Jens Sauer, and Denis Lalanne, "Introducing the recola multimodal corpus of remote collaborative and affective interactions," in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*. IEEE, 2013, pp. 1–8.

[19] Davis E King, "Dlib-ml: A machine learning toolkit," *The Journal of Machine Learning Research*, vol. 10, pp. 1755–1758, 2009.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097–1105.

[21] Xiao Zhang, Lei Zhang, Xin-Jing Wang, and Heung-Yeung Shum, "Finding celebrities in billions of web images," *Multimedia, IEEE Transactions on*, vol. 14, no. 4, pp. 995–1007, 2012.

[22] Hong-Wei Ng and Stefan Winkler, "A data-driven approach to cleaning large face datasets," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 343–347.