A Deep Feature based Multi-kernel Learning Approach for Video Emotion Recognition

Wei Li
Dept of Electrical Engineering
CUNY City College
New York, USA
lwei000@citymail.cuny.edu

Farnaz Abtahi
Dept of Computer Science
CUNY Graduate Center
New York, USA
fabtahi@gradcenter.cuny.edu

Zhigang Zhu
Dept of Computer Science
CUNY Graduate Center and
City College. New York, USA
zhu@cs.ccny.cuny.edu

ABSTRACT

In this paper, we describe our proposed approach for participating in the Third Emotion Recognition in the Wild Challenge (EmotiW 2015). We focus on the sub-challenge of Audio-Video Based Emotion Recognition using the AFEW dataset. The AFEW dataset consists of 7 emotion groups corresponding to the 7 basic emotions. Each group includes multiple videos from movie clips with people acting a certain emotion. In our approach, we extract LBP-TOP-based video features, openEAR energy/spectral-based audio features, and CNN (convolutional neural network) based deep image features by fine-tuning a pre-trained model with extra emotion images from the web. For each type of features, we run an SVM grid search to find the best RBF kernel. Then multi-kernel learning is employed to combine the RBF kernels to accomplish the feature fusion and generate a fused RBF kernel. Running multi-class SVM classification, we achieve a 45.23% test accuracy on the AFEW dataset. We then apply a decision optimization method to adjust the label distribution closer to the ground truth, by setting offsets for some of the classifiers' prediction confidence score. By applying this modification, the test accuracy increases to 50.46%, which is a significant improvement comparing to the baseline accuracy 39.33% .

Categories and Subject Descriptors

I.4.9 [**Image Processing and Computer Vision**]: Applications; I.5.4 [**Pattern Recognition**]: Applications

Keywords

Emotion recognition; multimodal features; deep learning; multi kernel learning

1. INTRODUCTION

Emotion recognition, which aims to obtain the type of emotion from captured data, which is either an image or a video clip, has been an interesting research topic for decades. In most existing works, emotions are classified into seven categories: Anger, Disgust, Fear, Happiness, Neutral, Sadness and Surprise. In this paper, we will use the same seven basic emotions. Several emotion datasets are available and many approaches have been applied to these existing datasets [20, 26].The EmotiW challenges in the past few years also followed the same categories of emotions. In the EmotiW 2015 challenge [8, 11], the AFEW5.0 is released by the organizers. The AFEW dataset contains short audio-video clips labeled with the above seven categories in both training and validation datasets. This challenge is a continuation of the EmotiW 2013 and 2014 [9, 10] challenges and the task is to assign an emotion label to the video clips from the unlabeled test dataset that is also provided by the organizers. Comparing to popular datasets, there are no restrictions on the faces in the EmotiW videos. The unconstrained lighting conditions, face poses and image qualities make the emotion recognition much more challenging. As shown in Figure 1, in some of the frames, detecting the face itself is also a challenge.

We propose a multi-feature fusion-based approach to tackle the EmotiW 2015 challenge. The approaches for emotion recognition on common datasets and the methods presented during the previous challenges inspired us to use a combination of engineered and learned features. In our approach, we first extract multimodal features from the images and videos, which include LBP-TOP features from videos, generating audio features from openEAR and extracting two CNN-based features from the fine-tuned model in two levels. Then we build SVM kernels using grid search and fuse the kernels by using Multi-Kernel Learning (MKL). We apply multi-class SVM classification to compute the fused kernel, then optimize the multi-class decision rules to obtain the final result.

The four main ideas proposed in our work are as follows. (1) Hybrid features - two engineered features and two deep-learning features. The two engineered features include LBP-TOP features [29] and openEAR-based audio features [12]. The feature selection rules are defined artificially. We further use CNN-based deep learning to extract learned features from image frames. (2) Multimodal features - video (spatial-temporal), audio (temporal), and image (spatial) deep features. We extract our features from multiple aspects. Since a video contains both temporal and spatial information, we try to take advantage of both. The LBP-TOP features take both individual and consequitive frames into consideration, obtaining both temporal and spatial in-

ICMI 2015, November 9–13, 2015, Seattle, WA, USA.

DOI: http://dx.doi.org/10.1145/2818346.2830583

Figure 1: Examples of challenging data.

formation. The audio conveys temporal information in a different modality, and the CNN features contain rich spatial information from the image. (3) The fine-tuning of the pre-trained deep learning model for extracting the two-level deep features. We propose using fine-tuning approach to solve the problem of insufficient images for training a deep CNN structure [13]. We apply our emotion data to train a pertained model for ImageNet [7] and obtain a fine-tuned structure. The fine-tuned model help us extract freatures from two levels of the CNN structure. (4) Multi-class SVM decision optimization to increase the overall emotion classification accuracy. We use multi-class SVM and also dig into the multi-class decision rules and figure out a decision score adjusting method to increase the overall accuracy.

The rest of the paper is organized as follows. In the next section, we review some of the most influential methods of emotion detection which are related to our proposed approach. Section 3 introduces the details of our method. The experiments and the results are explained in Section 4. Finally, the conclusions of our approach are presented in Section 5.

2. RELATED WORK

Most existing facial emotion recognition approaches have focused on recognizing emotions of frontal faces, such as the images in CK+ [20]. Shan, et al [22], proposed a LBP-based feature extractor combined with an SVM for classification. In the method proposed by Xiao, et al [28], comparing to training one model for all emotions, a separate model is trained for each emotion, which improves the performance. Wang, et al [27] modeled facial emotion as a complex activity that consists of temporally overlapping sequence of face events. Then, an Interval Temporal Bayesian Network (ITBN) was used to capture the complex temporal information. Zhao, et al [29] proposed LBP-TOP features to capture the dynamics of the video by transforming the video frames to an x-y-t cube and extracting LBP features from the x-t and y-t planes. The results on the CK+ dataset showed that these features are effective in recognizing emotions in videos.

During recent years, the vision community has been exposed to the wide spread of Deep Learning [25, 24]. Deep learning approaches are also used in emotion detection in many applications. Liu, et al [19] proposed a Boosted Deep Belief Network to perform feature learning, feature selection and classifier construction for emotion recognition. Different DBN models for unsupervised feature learning in audio-visual emotion recognition have been compared in the work done by Kim, et al [15]. Li, et al [17] used CNNs on images collected from the web. To prove the effectiveness of CNNs, they compared their performance on CK+ to the state of the art methods.

Many interesting approaches have been proposed in the EmotiW challenge. Karan, et al [23] extracted multiple channels of features such as Bag of Word (BoW), Histogram of Oriented Gradient (HOG), GIST, etc. To make all these features contribute to the final prediction, Multi-Kernel Learning (MKL) [2] was employed. The results showed that this feature fusion method improves the accuracy. A similar method is proposed by Chen, et al [6]. Liu, et al [18] also suggested a feature fusion approach for the challenge, which used Riemannian manifold to model the video features. Kahou, et al [14] trained a CNN-based on images obtained from the web and then used the class probabilities as deep features. DBNs were used to get valuable features from the audio channel. Afterwards, a random search algorithm [4] was applied, which achieved promising recognition rates.

3. THE PROPOSED APPROACH

The pipeline of our proposed approach includes 4 major parts: the hybrid multimodal features, the fine-tuning of a pre-trained CNN model for extracting deep image features, the multi-kernel learning for feature fusion, and the SVM-based multi-class emotion recognition structure with decision optimization. We will describe each part in the following subsections.

3.1 Hybrid multimodal features

The multimodal features that we utilize include the following three types: LBP-TOP-based video features, openEAR energy/spectral-based audio features, and CNN-based deep image features. The video and audio features convey spatial and temporal information. The learned Deep features focus more on emotion representation in (spatial) images. Our hypothesis is that the combination of both engineered (video and audio features) and learned features (CNNs features) can make the approach more robust.

3.1.1 LBP-TOP video features

Local Binary Pattern (LBP) has been effectively used in many computer vision applications. LBP can be used to describe image appearance by comparing local nearby pixels and generated patterns. A video can be described as a set of consecutive frames. Extracting LBP features from each frame and concatenating the features for the entire video results in lots of redundant information and the temporal characteristics of the video are also lost. Zhao, et al [29] proposed the LBP-TOP (LBP of Three Orthogonal Planes) features. Instead of extracting LBP features from the video frame by frame, we regard the video as a cube with x-y-t coordinates, where the x-y plane represent the spatial image

plane and the t axis corresponds to the time. The time axis contains rich temporal information. We extract LBP features on all three planes: x-y, x-t and y-t.

3.1.2 OpenEAR audio features

The speed or strength of the voice during speech can be useful indicators of different emotions. Emotions can be conveyed by voice through changes in pitch, loudness, timbre, speech rate, and pauses which is different from linguistic and semantic information. Study shows that anger and sadness are perceived most easily by using audio information, followed by fear and happiness [3]. To extract audio features, the openEAR tool is used. The tool can provide audio information including energy/spectral low-level features and also voice related features. These features form vectors of length 1583.

3.1.3 CNN-based deep image features

CNN-based deep learning is proved to be effective in image-based classification [16]. CNNs can be used to classify the images directly, or as a feature extractor to generate image features for other classifiers [25]. The CNN features are extracted through different CNN layers. Kahou [14] proposed to use CNN's predicted probabilities for 7 emotions as features, which is obtained from the last layer of the CNN. For videos, features of multiple frames are concatenated to form video features. In our approach, we also extract the probability distribution of the seven emotions from images of a video clip. Since we believe that the penultimate layer features may convey more useful information, we extract features from the penultimate layer as well. To make it simple, we call the probability features CNN-1 and the penultimate features CNN-2. In order to extract deep features from the CNN that are both robust and representative to the facial experession images, in the absence of sufficient number of facial images for training, we propose a unique fine-tuning method by using extra emotion images to update a pre-trained model. We will give more details about the fine-tuning method in the following subsection.

3.2 Fine-tuning a pre-trained model for deep image features

A simple-structured CNN trained on a small dataset is unable to learn the features deeply. Fine-tuning a pre-trained model with extra data can solve this problem. AlexNet is designed for ImageNet classification and shows very good performance [16]. We use AlexNet as our pre-trained model. Then, we need to fine-tune this model by training it on extra emotion images. Li [17] constructed the CIFE dataset by collecting more than 10 thousand instances of the 7 emotion classes. The number of samples of different emotions in the CIFE dataset are: Anger (1785), Disgust (266), Fear (781), Happiness (3636), Neutral (644), Sadness(2485) and Surprise(997). The images are from the web and most of them are not posed. The author reported a 83% recognition rate. This is the dataset we have used in our previous work. Since the number of samples in different classes are not balanced in CIFE, we have added some images to classes with fewer samples (for example Disgust and Fear) to balance the class sizes. After this modification, the number of samples for the 7 emotions become: Anger (1905), Disgust (975), Fear (1381), Happiness (3636), Neutral (2381), Sadness(2485) and Surprise(1993). As our first attempt, we have tried to see if we could use the same three-layer CNN structure as used by Li [17] to extract the learned features. However, with the same structure, the accuracy dropped from 83% to 65% with the balanced dataset. We have also observed the original high performance of 83% in Li's work was due to the unbalanced dataset: the recognition rates for disgust and fear classes were very low, for both the original dataset and the balanced dataset, and hence adding new samples decreased the overall performance. The reason for the low performance is that the three-layer structure is unable to learn the features deeply enough compared to the complicated structure of AlexNet for ImageNet test. Therefore, we choose to utilize the deeper structure of AlexNet as shown in Figure 2.

In the AlexNet structure, there are 5 convolutional layers, 3 fully connected layers, and 60 million parameter in total. Our first guess was that training the AlexNet on our data would results in better classification accuracy. The only problem was the need for larger number of images, as the ImageNet requires millions of images during training. Therefore, we instead propose a CNN fine-tuning method to train a deeper model based on AlexNet. The rationale is that although our task is different from the ImageNet, which focuses on object classification, similar low level filters could be used in emotion recognition. Based on this hypothesis, we can use the AlexNet and utilize our relatively 'small' dataset to update and fine-tune parts of its parameters for adapting it to emotion recognition.

As shown in Figure 2, the parameters of the convolutional layers 1 through 4 are not changed. Our new dataset is used to update the parameters of the convolutional layer 5 and the first fully connect layer, without changing their structures. In the original AlexNet, the number of units of the second fully connected layer and the third layer are 4096 and 1000 respectively. Since the number of classes in our dataset is just 7, we needed to change the structure in these two layers. We reduced the number of neurons in the penultimate layer to 256, and the third fully connected layer to 7. To train and fine-tune the CNN model, we set the learning rate to 0.01 and the training batch size to 256 images. The number of training iterations is chosen to be 800. The classification accuracy by using this model is 78.9% on the balanced CIFE dataset, which shows that the fine-tuning leads to a much better performance than our first attempt of using a three-layer CNN structure.

After the fine-tuning, the model is trained and tested on additional images from the web and then fine-tuned again using the training face images provided for the EmotiW 2015 challenge. By fine-tuning the CNN model, we obtain a model which works well on web images and is hopefully suitable for the challenge faces as well. For utilizing the model, deep image features are extracted from the CNN. Note that the input data that we need to process include videos with different lengths. In order to obtain normalized features, we need to make the number of samples equal in all videos. In our approach, we decided to set the number of samples for all videos to 32, experimentally. Hence, we need to downsample the frames for videos with more than 32 samples and interpolate for those videos with fewer than 32 samples.

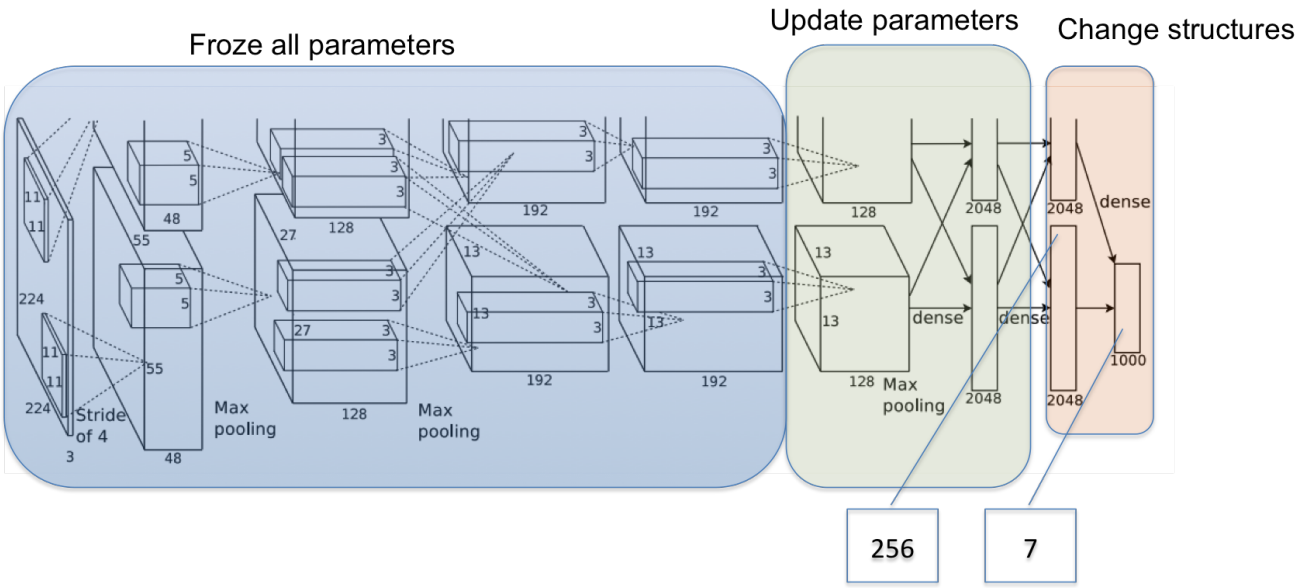

Figure 2: Fine-tuning of the Alexnet

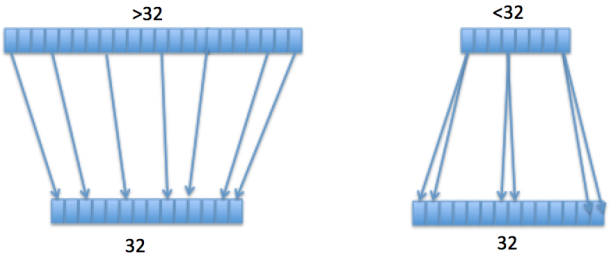

Figure 3: Normalizing the video frames

This process is illustrated in Figure 3. Then we use our fine-tuned CNN model to extract the features from the "normalized" videos. Unlike the work done by Kahou [14] where only the probability vector of the 7 emotions was used, we also extract the penultimate layer features. So we can obtain 2 vectors of lengths 32x7=224 and 32x256=8192, respectively.

3.3 Multiple kernel learning

So far we have explained how we obtain the four types of features we intend to use: LBP-TOP from the videos, openEAR from the audio, and CNN-1 and CNN-2 from the images. Now the next question is how to effectively combine these features to achieve better a emotion recognition rate. Similar to the work done in [23], we decide to use MKL+SVM: multi-kernel learning with SVMs [21, 5, 2]. SVM is a well-known and widely used classifier. The goal of the SVM classifier is to find a hyperplane through the objective function in Equation 1.

$$max[\sum_{i=1}^{N} \boldsymbol{a}_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} \boldsymbol{a}_i \boldsymbol{a}_j y_i y_j K(x_i, x_j)] \quad (1)$$

with

$$\sum_{i=1}^{N} \boldsymbol{a}_i y_i = 0, 0 \leq \boldsymbol{a}_i \leq C \quad (2)$$

where x represents the sample and y represents the corresponding label, which is either 1 or -1. C is the penalty factor. K is the SVM kernel, which employs a kernel function (such as RBF in Equation 5 in Section 4) to encode the samples to generate an NxN matrix, where N is the number of samples. The a parameters can be learned by optimization which results in a binary SVM classifier.

The MKL+SVM method is based on SVM, where the only difference is the kernel. The kernel in MKL is a linear combination of SVM kernels as described in Equation 3.

$$K_{mkl}(x_i, x_j) = \sum_{m=1}^{M} \boldsymbol{b}_m K_m(x_i, x_j) \quad (3)$$

with

$$\boldsymbol{b}_m \geq 0, \sum_{m=1}^{m} \boldsymbol{b}_m = 1 \quad (4)$$

where b is the coefficient of the SVM kernels Km *(m=1, ..., M)*. So, each channel of features will generate an SVM kernel, then MKL finds the best coefficients to linearly combine these individual kernels. In this work, the type of the kernels are chosen to be Radius Basic Functions (RBFs).

3.4 SVM-based system structure

SVMs are initially used for binary classification. In our approach, we use multiple binary SVM classifiers to accomplish the classification task of 7 emotions. In many applications,

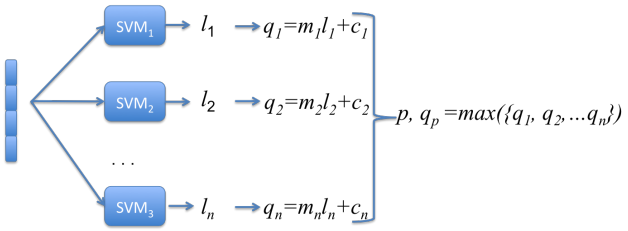

Figure 4: Framework for multi-class SVM prediction

SVM-based multi-class classification is realized by generating and integrating multiple binary SVM models. For an n-class problem, n binary SVM models are trained individually. Then, the confidence (l) of each sample belonging to a specific class is computed by the corresponding SVM. These n confidence values then need to be combined. The easiest and most commonly used way for combining the results is to pick the maximum score and assign the corresponding label to the sample. This may not be the best method to perform multi-class classification, especially when the binary classifiers' accuracies are different due to different number of positive samples used to train them. We propose to solve this problem either by giving a weight (m) to each classifier, or by setting offsets (c) for them, as shown in figure 4, to optimize the recognition results based on the training data. We will explain this method in more details later in section 4.2.

4. EXPERIMENT

In this section, we will discuss the critical components in boosting up the accuracy of emotion recognition with challenging video clips. These components include the construction of the hybrid features from both engineered and learned features, and the optimization of the final classification decisions based on the results of the seven SVM-based classifiers. Then we will discuss our experimental results, emphasizing the role of these two important steps.

4.1 Constructing the features

As discussed earlier, MKL is applied to the RBF kernels. Thus, before using MKL, we need to transform the extracted features to RBF kernels. There are two parameters for the RBF-based SVM that need to be determined: The penalty factor C and the parameter r which appears in the RBF Equation 5.

$$K(x, x') = exp(-r||x - x'||) \tag{5}$$

All SVM classifiers will share the same penalty factor. To make it easy to compute and combine the kernels, we set the penalty factor to the default value 1. For determining r, we perform a grid search for each feature and evaluate the performance of the corresponding SVM on the validation dataset. To achieve the goal of multi-class classification, we use a one-vs-all scheme for each of the emotion classes. This will yield seven classification scores from -1 to 1 for each sample. The scores express the confidence of each classifier in assigning the corresponding class label to that particular sample. For each sample, the class corresponding to the maximum confidence is selected.

Table 1: Comparison of the classification accuracy of different features on each emotion (L-T represents LBP-TOP)

	A	D	F	H	N	Sa	Su	r
L-T	0.75	0	0.04	0.74	0.61	0.17	0.17	0.1
Audio	0.74	0	0.32	0.32	0.52	0.22	0.02	0.1
CNN-1	0.47	0.12	0.15	0.65	0.34	0.32	0.13	0.1
CNN-2	0.58	0.15	0.13	0.67	0.54	0.42	0.23	0.0001

As mentioned before, the best r values for generating the RBF kernels is determined by a grid search. Table 1 shows the r parameter as well as the accuracy for each feature. In this table, we use A, D, F, H, N, Sa, Su to represents Anger, Disgust, Fear, Happiness, Neutral, Sadness and Surprise, respectively. The video features perform very well in recognizing anger, happiness and neutral, but terrible in fear, disgust and sadness. The audio features basically follow the same pattern, but work best in fear. For the deep features, CNN-2 almost outperforms CNN-1 in all classifiers. In the emotions where the video features work best, the CNN-1 does not demonstrate high accuracy, but in emotions like sadness and surprise, the CNN-2 performs much better. The difference in performance could be due to the different focuses of learned and engineered features. The LBP based features represent the facial "texture" well, so for expressions like happy, angry and neutral, the difference in texture is easier for the LBP features to capture and represent. On the other hand, the deep learning features are based on the comparison of different samples and minimizing the loss. So, these features not only capture lower level characteristics like texture, but also some higher level information that differentiate similar emotions. These observations actually draw our attention towards the use of MKL.

we compute coefficients for the kernels by running the MKL algorithm [5]. The coefficients found for video, audio, CNN-1 and CNN-2 features are 0.56, 0.08, 0.0, 0.36. It make sense that the coefficient of CNN-1 is 0 since CNN-2 performs better in most classes. This is also a strong evidence of the effectiveness of extracting CNN-2. Having the coefficients, we can construct the combined kernel and test it on the validation data. By applying the multi-class SVM classification using the fused kernel, we achieve an accuracy of 48.52% on the validation dataset. The confusion matrix of the combined kernel on the validation dataset is shown in Table 2. Compared to the baseline approach that were provided by the organizers, our approach shows a 9% increase in the accuracy. This is the best performance we achieved on the validation data before we use decision optimization; the model also achieves 45.23% accuracy when applied to the test dataset, which is more accurate compared to the 39.33% accuracy of the baseline approach [1].

4.2 Decision optimization

As mentions in Section 3.4, we can use different methods to decide the final classification results in a multi-class SVM. In our initial approach, classification is performed by com-

Table 2: The confusion matrix of the best validation result

	A	D	F	H	N	Sa	Su
A	0.72	0.03	0.05	0.05	0.05	0.05	0.03
D	0.25	0.17	0.05	0.17	0.23	0.05	0.05
F	0.34	0.02	0.13	0.11	0.18	0.02	0.18
H	0.07	0.01	0.03	0.74	0.09	0.01	0.01
N	0.03	0.06	0.01	0.11	0.67	0.06	0.03
Sa	0.13	0.11	0.01	0.08	0.22	0.33	0.08
Su	0.17	0.04	0.08	0.10	0.30	0.02	0.26

Table 3: Test confusion matrix

	A	D	F	H	N	Sa	Su
A	0.73	0.01	0.07	0.02	0.08	0.03	0.02
D	0.07	0.03	0.10	0.20	0.10	0.34	0.14
F	0.36	0.02	0.16	0.07	0.13	0.15	0.10
H	0.11	0.01	0.03	0.56	0.12	0.10	0.06
N	0.12	0.03	0.07	0.09	0.46	0.15	0.06
Sa	0.21	0.01	0.03	0.15	0.04	0.50	0.04
Su	0.18	0.07	0.14	0.07	0.11	0.26	0.15

paring all seven confidence probabilities and selecting the index with the maximum probability value as the final label. A shortcoming of this approach is that some classifiers are weaker as there are unbalanced number of samples across the seven categories in the training dataset. So the output of those classifiers tend to have lower confidence in classification, which may affect the final classification accuracy.

Table 3 shows the sample distribution in the test dataset across the seven classes estimated from the test accuracy for our first submission in Section 4.1, shown in row 1 as "Ground-truth", even though we don't know the ground truth labels. It is obvious that the test dataset is very unbalanced and for some of the emotions, the number of samples are either too low (surprise) or two high (neutral). This leads us to the idea of modifying the decision rule by giving higher scores to under-estimated classes. For instance, we can increase the score for the neutral classifier, as we expect to have more samples classified to that class. This is similar to the use of priors in classification, like reinforcement learning to adjust decision making based on feedback. Now we need to determine the amount of change that we need to apply to the scores.

We assume that if the distribution of the estimated classes is similar to the "ground truth" class distribution, which could be obtained from previous experience, we may have

Table 4: The number of samples for different emotions

	A	D	F	H	N	Sa	Su
Ground truth	79	29	66	108	159	71	27
Before optimization	136	15	40	99	109	102	38
After optimization	114	3	20	97	209	78	18

Table 5: Test confusion matrix after optimization

	A	D	F	H	N	Sa	Su
A	0.69	0	0.13	0.03	0.22	0.05	0
D	0.07	0	0.07	0.21	0.21	0.34	0.10
F	0.32	0.02	0.12	0.07	0.32	0.09	0.06
H	0.08	0	0.09	0.58	0.26	0.03	0.03
N	0.06	0.13	0.19	0.07	0.69	0.11	0.03
Sa	0.15	0	0.03	0.13	0.20	0.48	0.01
Su	0.18	0	0.11	0.04	0.44	0.11	0.11

the chance to obtain higher accuracy by adjusting the decision rules. So in the decision level, we give offsets to some classifiers, that is, we increase the score of those classes that are assumed to be more "popular" due to their higher number of samples. The distribution of the samples to the seven classes estimated by our model before decision optimization is shown in row 2 of Table 3. Based on Table 3 (first 2 rows), we apply this "optimization" to two classes, neutral and happiness. By trying different values for the offsets, we found that increasing the weights by 0.03 and 0.1 will bring the class distribution curve closer to the ground truth. This can be seen in the last row of Table 3. Since we do not know the ground truth, we can not be completely sure that we are using the best pair of offsets. We believe that the values of the offsets can further be optimized. The decision optimization changes our final overall test accuracy to 50.24%, more than 10% improvement over the baseline approach [1]. Our validation accuracy after decision optimization drops to 46.7%. This is understandable since the validation dataset is pretty balanced in the number of samples across the classes. The confusion matrices on the test dataset before and after decision optimization are shown in Table 4 and Table 5, respectively. By comparing the two confusion matrices, we can see that there is an increase in the accuracy of neutral and happy classifiers and a slight drop in the other classifiers. But overall, we have a significant improvement for the test dataset.

4.3 Discussions

By employing the MKL method and decision level optimization, we obtain the accuracy of 50.46%. Compared to the baseline provided by the organizers, the improvement is significant. We have three main contributions in our work:

1. Deep image features are obtained using the AlexNet model. The features generated by AlexNet are very effective in object classification. In order to take advantage of this model, we proposed a fine-tuning method which keeps most of the convolutional filters unchanged, but modifies part of the structure of AlexNet. We apply a fine-tuning training method on a dataset of images obtained from the web, plus the challenge face images to adapt AlexNet to the challenge.

2. Mutli-kernel learning is used to fuse both the engineered and learned features from different channels. These different features (video, audio and images) emphasize on different characteristics. Optimized coefficients of the MKL enables the combined classifier to have the benefit of all individual classifiers and therefore achieves a better overall performance.

3. We utilize a decision optimization method for SVM-based multi-class classification. Instead of treating the confidences of all classifiers equally and assigning the class label corresponding to the highest confidence score, we adjust the decision mechanism by adding extra confidence to the classes that are expected to have more samples classified correctly. This brings the predicted distribution of the classes closer to the ground truth. This may lead to some decrease in the number of predicted samples belonging to some of the classes, but overall, the classification accuracy is improved.

5. CONCLUSION

In this paper, we described the approach that we have proposed and tested to participate in the EmotiW 2015 challenge. Our method combines hybrid and multimodal features for emotion classification, which cover various channels - video (spatial-temporal), audio (temporal) and image (spatial), and include both engineered (LBP-TOP, openEAR), and learned (CNN) features. The CNN features are obtained from our fine-tuned model which is trained on additional images from the web and aligned face images provided by the challenge. Two levels of CNN features are learned. The features are combined using MKL by generating an RBF kernel that is a weighted combination of individual kernels from each classifier. It turned out that the CNN features from the penultimate layer are more effective than the previously used CNN features (probability feature from the last layer). After analyzing the distribution of the classification results, we optimized the multi-class decision rules to make the distribution of the predicted classes similar to the ground truth data. The accuracy of classification after applying this optimization is 50.46% which is significantly higher than the accuracy of the baseline method.

6. ACKOWLEDGEMENT

This work is supported in part by the National Science Foundation (Award # EFRI -1137172), VentureWell (formerly NCIIA, through Award # 10087-12), and a CUNY Graduate Center Enhanced Chancellor Fellowship (to Abtahi).

7. REFERENCES

[1] D. Abhinav, M. O.V. Ramana, G. Roland, J. Jyoti, and G. Tom. Video and image based emotion recognition challenges in the wild: Emotiw 2015. In *ICMI*, 2015.

[2] F. R. Bach, G. R. Lanckriet, and M. I. Jordan. Multiple kernel learning, conic duality, and the smo algorithm. In *Proceedings of the twenty-first international conference on Machine learning*, page 6. ACM, 2004.

[3] J.-A. Bachorowski. Vocal expression and perception of emotion. *Current directions in psychological science*, 8(2):53–57, 1999.

[4] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *The Journal of Machine Learning Research*, 13(1):281–305, 2012.

[5] S. Bucak, R. Jin, and A. K. Jain. Multi-label multiple kernel learning by stochastic approximation: Application to visual object recognition. In *Advances in Neural Information Processing Systems*, pages 325–333, 2010.

[6] J. Chen, Z. Chen, Z. Chi, and H. Fu. Emotion recognition in the wild with feature fusion and multiple kernel learning. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 508–513. ACM, 2014.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE, 2009.

[8] A. Dhall et al. Collecting large, richly annotated facial-expression databases from movies. 2012.

[9] A. Dhall, R. Goecke, J. Joshi, K. Sikka, and T. Gedeon. Emotion recognition in the wild challenge 2014: Baseline, data and protocol. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 461–466. ACM, 2014.

[10] A. Dhall, R. Goecke, J. Joshi, M. Wagner, and T. Gedeon. Emotion recognition in the wild challenge 2013. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 509–516. ACM, 2013.

[11] A. Dhall, R. Goecke, S. Lucey, and T. Gedeon. Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark. In *Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on*, pages 2106–2112. IEEE, 2011.

[12] F. Eyben, M. Wöllmer, and B. Schuller. Openear—introducing the munich open-source emotion and affect recognition toolkit. In *Affective Computing and Intelligent Interaction and Workshops, 2009. ACII 2009. 3rd International Conference on*, pages 1–6. IEEE, 2009.

[13] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the ACM International Conference on Multimedia*, pages 675–678. ACM, 2014.

[14] S. E. Kahou, X. Bouthillier, P. Lamblin, C. Gulcehre, V. Michalski, K. Konda, S. Jean, P. Froumenty, A. Courville, P. Vincent, et al. Emonets: Multimodal deep learning approaches for emotion recognition in video. *arXiv preprint arXiv:1503.01800*, 2015.

[15] Y. Kim, H. Lee, and E. M. Provost. Deep learning for robust feature generation in audiovisual emotion recognition. In *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pages 3687–3691. IEEE, 2013.

[16] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[17] W. Li, M. Li, Z. Su, and Z. Zhu. A deep-learning approach to facial expression recognition with candid images. In *Machine Vision Applications (MVA), 2015 14th IAPR International Conference on*, pages 279–282. IEEE, 2015.

[18] M. Liu, R. Wang, S. Li, S. Shan, Z. Huang, and X. Chen. Combining multiple kernel methods on riemannian manifold for emotion recognition in the

wild. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 494–501. ACM, 2014.

[19] P. Liu, S. Han, Z. Meng, and Y. Tong. Facial expression recognition via a boosted deep belief network. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1805–1812. IEEE, 2014.

[20] P. Lucey, J. F. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews. The extended cohn-kanade dataset (ck+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010.

[21] A. Rakotomamonjy, F. Bach, S. Canu, and Y. Grandvalet. Simplemkl. *Journal of Machine Learning Research*, 9:2491–2521, 2008.

[22] C. Shan, S. Gong, and P. W. McOwan. Facial expression recognition based on local binary patterns: A comprehensive study. *Image and Vision Computing*, 27(6):803–816, 2009.

[23] K. Sikka, K. Dykstra, S. Sathyanarayana, G. Littlewort, and M. Bartlett. Multiple kernel learning for emotion recognition in the wild. In *Proceedings of the 15th ACM on International conference on multimodal interaction*, pages 517–524. ACM, 2013.

[24] Y. Sun, X. Wang, and X. Tang. Deep learning face representation from predicting 10,000 classes. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1891–1898. IEEE, 2014.

[25] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pages 1701–1708. IEEE, 2014.

[26] M. Valstar and M. 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*, page 65, 2010.

[27] Z. Wang, S. Wang, and Q. Ji. Capturing complex spatio-temporal relations among facial muscles for facial expression recognition. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 3422–3429. IEEE, 2013.

[28] R. Xiao, Q. Zhao, D. Zhang, and P. Shi. Facial expression recognition on multiple manifolds. *Pattern Recognition*, 44(1):107–116, 2011.

[29] G. Zhao and M. Pietikainen. Dynamic texture recognition using local binary patterns with an application to facial expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 29(6):915–928, 2007.