Combining Modality Specific Deep Neural Networks for Emotion Recognition in Video

ICMI 2013, EmotiW 2013 Grand Challenge

Samira Ebrahimi Kahou (samira.ebrahimi.kahou@gmail.com)

with : **Christopher Pal**, Xavier Bouthillier, Pierre Froumenty, Çaglar Gülçehre, Roland Memisevic, Pascal Vincent, Aaron Courville, Yoshua Bengio and …

POLYTECHNIQUE MONTRÉAL
WORLD-CLASS ENGINEERING

Université de Montréal

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

McGill

Sydney, Australia. December 9-13th, 2013

Additional Authors : ***LISA - Université de Montréal :*** Raul Chandias Ferrari, Mehdi Mirza, Sébastien Jean, Pierre-Luc Carrier, Yann Dauphin, Nicolas Boulanger-Lewandowski, Abhishek Aggarwal, Jeremie Zumer, Pascal Lamblin, Jean-Philippe Raymond, Guillaume Desjardins, Razvan Pascanu, David Warde-Farley, Atousa Torabi, Arjun Sharma, Emmanuel Bengio, and Myriam Côté. ***Goethe Universität - Frankfurt :*** Kishore Reddy Konda. ***McGill University :*** Zhenzhou Wu.

Acknowledgements : We thank NSERC, Ubisoft and the German BMBF, project 01GQ0841.

POLYTECHNIQUE MONTRÉAL
WORLD-CLASS ENGINEERING

Université de Montréal

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

McGill

Our Models and Combination Techniques

Models

1. Faces and Convolutional Network #1
   Yields higher performance on the challenge validation set
2. Faces and Convolutional Network #2
3. Deep Restricted Boltzmann machine based audio model
4. Deep Auto-encoder based technique for activity recognition
5. Shallow neural network model based on bag of mouth features

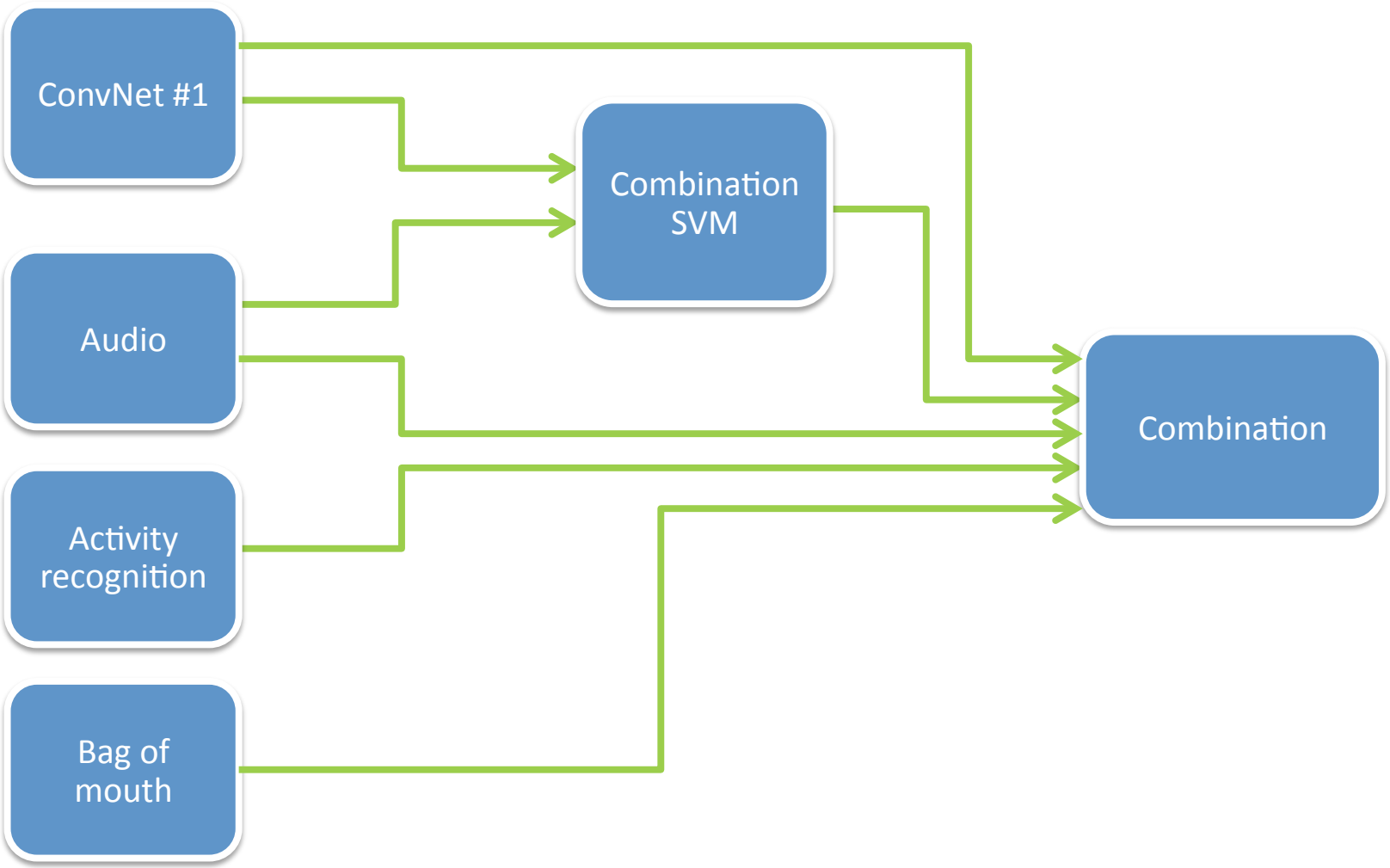
Combining Models

1. SVM combination of models
2. Flat Averaging of Models
3. Weighted Averages obtained via Random Search

Results

The challenge baseline : 27.56% Accuracy
Our best single model : 35.6%
Our best submission : 41.03%

Overview of the Approach Yielding the Win



Comparing Models with Confusion Matrices

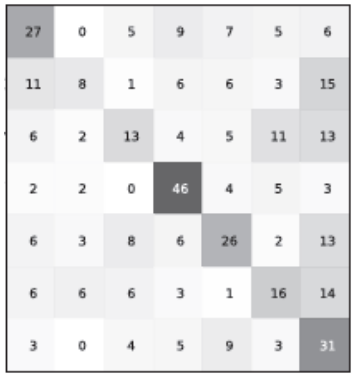
CN#1	CN#2	Audio	BoM	Act. Rec.
25 1 0 10 6 8 9	40 3 0 4 0 6 6	34 0 3 1 10 5 6	22 6 2 12 3 10 4	18 3 2 15 2 6 13
17 1 0 8 10 5 9	14 1 2 16 0 2 15	10 1 2 5 9 0 23	11 4 4 10 5 7 9	7 7 1 20 2 1 12
9 1 3 5 12 10 14	16 1 7 7 0 5 8	10 0 12 2 12 4 14	20 6 12 4 2 7 3	10 0 15 13 2 6 8
4 0 0 45 8 4 1	20 0 0 38 0 2 2	13 0 5 4 14 6 20	11 7 1 32 3 6 2	12 3 5 24 2 7 9
11 0 1 7 25 6 14	21 1 2 7 0 6 27	1 0 12 3 30 2 16	13 5 9 5 11 7 14	10 6 2 28 4 1 13
10 1 1 2 2 22 14	20 0 5 5 0 7 15	6 0 6 2 6 10 22	7 2 8 6 4 22 3	4 1 3 17 0 9 18
8 1 0 4 6 6 30	16 0 3 6 0 1 29	4 0 1 9 14 2 25	11 9 0 9 5 5 16	5 4 4 9 4 5 24
Train: 45.8	34.0	58.2	96.3	100.0
Valid: **38.1**	30.8	**29.3**	30.1	25.5
Test: **35.6**				

- Convolutional networks have similar error profiles
- Notice different error profiles for other model classes
- Tried using MLPs and SVMs to combine all models, but they overfit (Question: Why?)

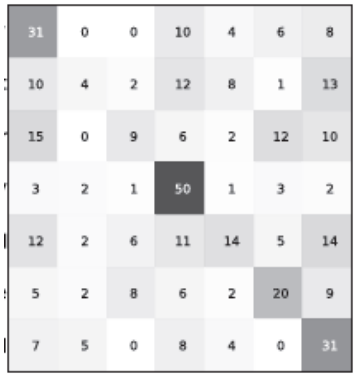
Combining Models

- BoM and Activity recognition techniques overfit the data
- Challenge: How could we exploit all the meaningful information provided by all models (including overfitters)

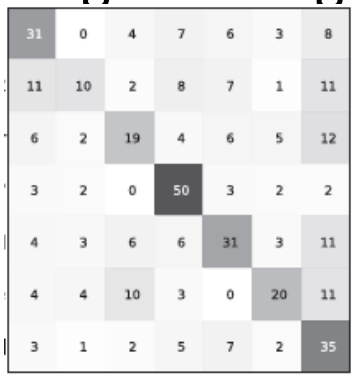
CN#1 & Audio



All, averaged



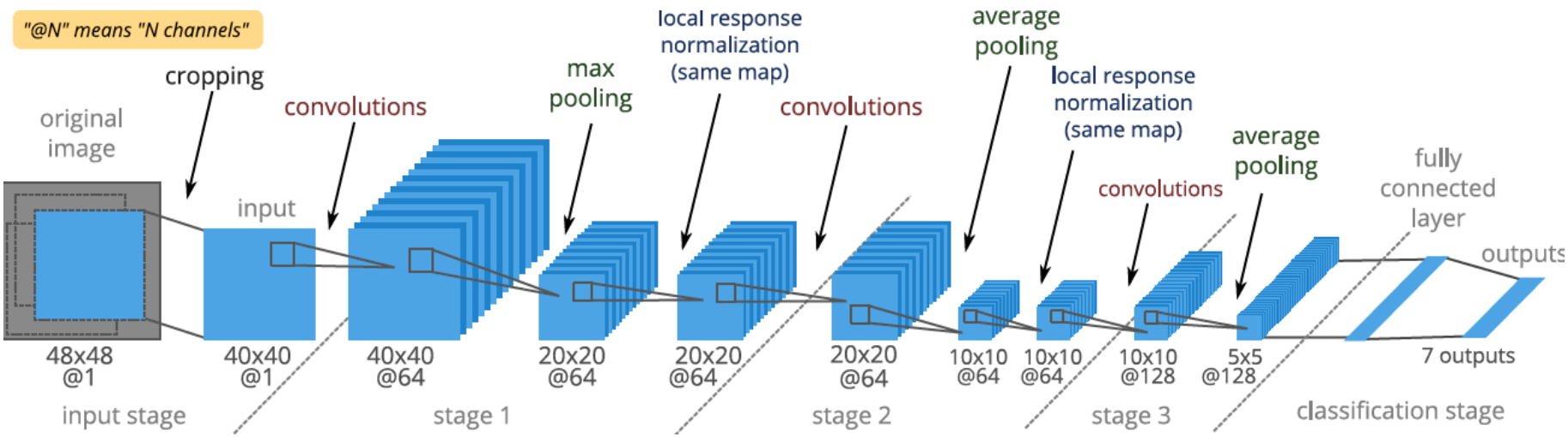
Random search, weighted avg.



	CN#1 & Audio	All, averaged	Random search, weighted avg.
Train:	71.8	97.1	92.4
Valid:	42.2	40.1	49.5
Test:	**38.5**	**37.1**	**41.0**

Faces and Convolutional Network #1

The Convolutional neural network architecture:



- Based on the C++ and Cuda implementation of Krizhevsky et al
- Inputs are images of size 40x40, cropped randomly
- Four layers, 3 convolutions followed by max or average pooling and a fully-connected layer

Why are Deep Neural Networks so Hot?

- Top results on the ImageNet contest
  (Krizhevsky, Sutskever & Hinton, 2012)
  1000 classes in 1.2 million images

  Deep Neural Network : 15% top-5 error rate
  Second best entry: 26% top-5 error rate

- Significant increase in performance for Large-Vocabulary Speech Recognition (Dahl, Yu, Deng & Acero, 2012)

  Deep Neural Network : 16-23% relative error rate reduction over the previous state-of-the-art (context-dependent Gaussian mixture model (GMM)-HMMs)

Key Aspect of Our Approach Using ConvNet #1

Our network is trained with two large static image databases of seven emotions

1. The Toronto Face Dataset (TFD)

- 4,178 images of frontal faces
- Pre-processed based on registering the eyes and then resized to 48x48

2. A Large facial expression database harvested from Google image search (created by us)

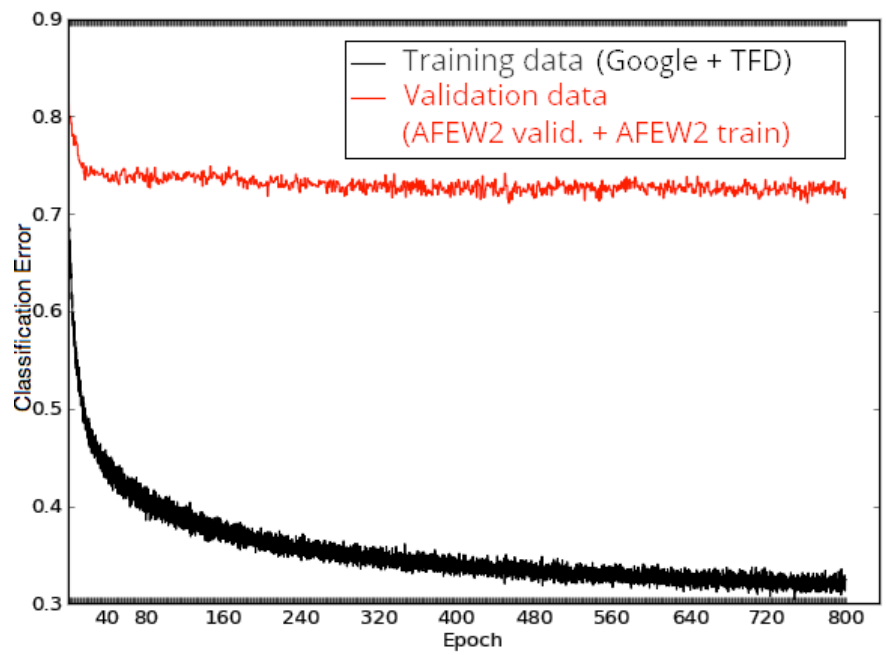
- 35,887 images with more pose variability
- Built by harvesting imagery returned from Google's image search using keywords related to expressions

Key Training & Aggregation Strategies for ConvNet #1

- Training the convolutional neural network on frames containing a face extracted from the video clips of the challenge training set plus the Google & TFD datasets: **96.73%** accuracy on training set and **35.32%** on the validation set

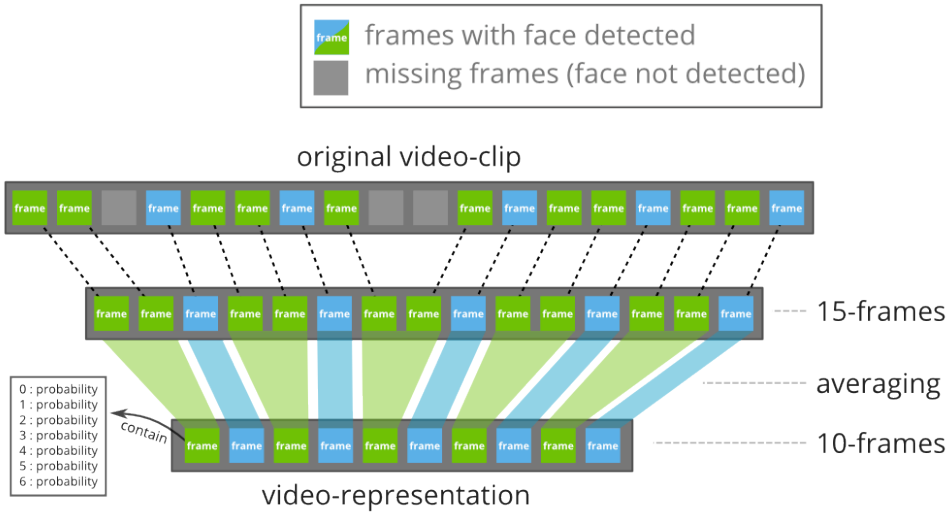
- Ignoring the Challenge training data, just using Google and the TFD to train the deep network, then training an SVM aggregator on the deep network predictions using the challenge training data: **46.87%** accuracy on train and **38.96%** on the validation set

ConvNet 1 classification error on training and validation sets (before aggregation)

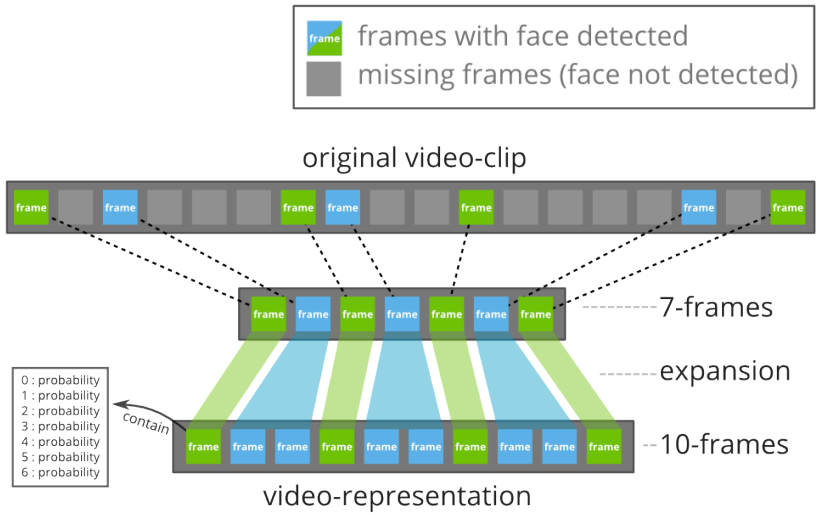


Our best network was therefore trained on the Google and TFD, using early stopping based on the challenge train and validation sets

ConvNet #1: Per frame prediction aggregation with an SVM



Averaging



Expansion

- Using ConvNet 1, we classified all frames from train, validation and test challenge data
- For each frame the output is a 7-class probability vector
- These probabilities are aggregated to build a fixed-length representation for each video-clip
- Finally, we trained an SVM with RBF kernel on the new representation for challenge train set and tuned hyper-parameters using validation set

Pre-processing for Convolutional Network #1

Facetube extraction procedure

- Frames were extracted in a way that the movie aspect ratio was preserved
- Google Picasa face detector was used to detect faces
- In order to recover bounding boxes for the faces returned by Picasa, we used Haar-like feature-based matching method as direct pixel to pixel matching did not yield satisfactory performance.

Registration

- The challenge dataset and TFD images are registered to the Google dataset using 51 facial keypoints extracted from Ramanan method
- To reduce the noise computed mean-shape with no-pose for each dataset
- Computed a similarity transformation between the two shapes
- We also added a random noisy border to TFD data

Illumination normalization

- We used a diffusion-based approach called Isotropic Smoothing (from INface toolbox)

Raw images at the top and their IS-preprocessed corresponding images below

Faces and Convolutional Network #2

- A simpler ConvNet, trained on 48x48 images from the TFD
- Pre-processed with local contrast normalization
- A single convolutional layer and a max pooling, followed by a hidden layer and a 7-class softmax output
- Faces from challenge video frames are aligned to have roughly the same eye positions as TFD using the average of eye related landmarks
- Each sequence was summarized using a few statistics such as : average, max, average of maximum suppression vectors
- Here, instead of SVM we used a multilayer perceptron trained on fixed-length feature vectors we have produced
- The MLP was composed of one rectified linear layer followed by a linear output of 7 units
- Early stopping based on validation set

Audio and Deep RBMs

Features

- Mel-frequency cepstral coefficients (MFCC) developed for speech recognition
- Background noise and the soundtrack of the movie can also be significant indicators
- Features are extracted from the mp3 files extracted from the movie clips using yafee library with a sampling rate of 48kHz
- Different types of MFCC features are used and 909 features per timescale were selected by online PCA

Pre-training

- Unsupervised pre-training with deep belief networks (DBN) on the extracted audio features
- DBN has two layers of RBMs. First layer RBM was a Gaussian RBM with noisy rectified linear unit (ReLU) nonlinearity. Second layer RBM was a Gaussian-Bernoulli RBM
- The MLP was initialized with <u>ReLU nonlinearity for 1st layer</u> and <u>sigmoid nonlinearity for 2nd layer</u> using the weights and biases of the DBN

Other Aspects of the Deep RBM Audio Model

Temporal pooling for audio classification

- The last hidden representation layer of an MLP was pooled to aggregate information across frames before a final softmax layer

Supervised fine-tuning

- Only challenge training data was used for fine-tuning and the early stopping was based on error rate on validation set

Activity Recognition and Deep Auto-encoders

- Based on spatio-temporal motion patterns in the video
- Recognition pipeline uses spatio-temporal auto-encoder for feature learning

- Model was trained on PCA-whitened input patches of size 10x16x16 cropped randomly from training videos
- A stride of 4 was used to generate 8 sub blocks from super blocks, then using PCA we built descriptors

- Using K-means on super block descriptors we learned a vocabulary of 3000 words
- Each video is represented as the histogram of these words and then classified using an SVM

Bag of Mouth Features and Shallow Networks

Uses aligned faces provided by the organizers

- Images are cropped to only keep a small region around the mouth
- The mouth images are subdivided into 16 regions, then many 8x8 patches are extracted from training images
- The patches are normalized and whitened keeping 90% of the variance
- A dictionary is learned for each of the 16 regions using K-means clustering

Learning

- For all images of train, patches are extracted densely
- Each patch is then represented by a 400 dimensional feature vector (number of clusters) by using triangle activation
- A frame by frame classifier using logistic regression is trained

Classification

- Probability of a clip is computed by averaging probability vectors of frames

Detailed Results

Our 7 submissions with training, validation and test accuracies

S	Train	Valid	Test	Method
7	92.37	49.49	**41.03**	Moderate local random search : Activity, Audio, Bag of mouth, CN1, CN1 + Audio
6	94.74	48.48	40.06	Short local random search : Activity, Audio, Bag of mouth, CN1, CN1 + Audio
5	94.74	47.98	39.42	Short uniform random search : Activity, Audio, Bag of mouth, CN1, CN1 + Audio
4	98.68	43.69	32.69	SVM with detailed hyper-param. search: Activity, Audio, Bag of mouth, ConvNet 1
3	97.11	40.15	37.17	Mean prediction from: Activity, Audio, Bag of mouth, ConvNet 1, ConvNet 2
2	71.84	42.17	38.46	ConvNet 1 (from submission 1) combined with Audio model using another SVM
1	45.79	38.13	35.58	Google data & TFD used to train ConvNet 1, frame scores aggregated with SVM

Main Contributions, Conclusions and Findings

- Using large quantities of additional data to train a deep convolutional neural network

  - Allows us to train high capacity model without over-fitting to the relatively small EmotiW challenge training data

- The strategy of using the challenge training data only to learn how to aggregate the per frame predictions

  - Highlights the issue of overfitting when using a relatively small amount of labeled video data

- A novel technique to aggregate (multimodal) models based on random search

  - Exploits the complementary information within the predictions of all models
  More robust in the presence of models that overfit

Thank You