# Automated Restyling of Human Portrait Based on Facial Expression Recognition and 3D Reconstruction

Cheng-Han Wu[1], Hsin Chen[2]
Department of Electrical Engineering[1] and Department of Mechanical Engineering[2], Stanford University
350 Serra Mall, Stanford, CA 94305, USA
[1]chw0208@stanford.edu
[2]hsinc@stanford.edu

## Abstract

*This project demonstrated an innovative automatic restyling system that turns a plane human portrait to one with effects that correspond to his/her facial expression. By training recognition models using convolutional neural network(CNN) and a modified classification algorithm, the system is able to detect emotion of the person in a picture. Based on the emotion, the system modifies the photo in a way that highlight the user's feeling.*

## 1. Introduction

*[Note: This project received permission from CS229 and Cheng-Han Wu's EE368: Digital Image Processing class to share the same base on the image processing part. The machine learning part of the project was developed jointly by the two authors.]*

Inspired by the movie Inside Out, the project showcased the capabilities of AI in visual effects automation in photography or film making. Using CNN with enhanced classification algorithms, we are able to achieve high expression recognition accuracy to be used with afterward visual effects. As of today, large amounts of video editing was done through post-editing with intense and costly human labor work. Since rendering of video or photos heavily depends on the actors' or models' expression, the project aim to perform post-rendering based on auto-perception of character emotions.

We experimented on two different neural network models with two different dataset sizes. After extracting features before the prediction layer of the neural network, we were able to test the performances of individual classification algorithms and see if we can substitute the existing softmax algorithm with better classification methods. After performing several tests with the algorithms with cross-validation and unseen data, we were able to construct a prototype of automatic restyling system based on expression recognition.

## 2. Related work

### 2.1. Machine Learning: Expression Recognition

Pattern recognition has long been a popular and growing field in computer vision. Current expression recognition researches have shown dramatic improvements in precision and efficiency. From the early feature line and eigenfaces detection[1], the task has been tackled using various approaches. The implementation of Gabor filtering detection in [2] gave rise to facial structural coding which gave a quantitative definition of expression.

Fasel et al. [3] demonstrated preliminary success in facial expression recognition using CNN. Using an ensemble of CNNs, Yu and Zhang [4] won EmotiW 2015 and achieved state-of-the-art results in facial expression recognition. The main boost in accuracy ($\sim$3%) came from adding random perturbation to the input images. During testing, they obtained multiple predictions of a single test image by randomly adding different perturbations to it, and voted on the produced labels to get a final answer.

Due to the limit of dataset quantity as well as subtle difference between certain emotions, high precision recognition is still an active research goal in this field.

### 2.2. Image Processing

Recent research of Matthias [5] demonstrated a way of capturing synchronized depth frames to perform character relighting of a movie clip. Depth information can also be used as masks to clearly distinguish targets in the scene specified by the range of depth. Research by Yang et. al. [6] demonstrated realistic relighting based on image morphing of facial images with a 3D facial model. The MRF-based method mentioned in the research was effective in facial relighting, delighting, and correction in extreme lighting condition.

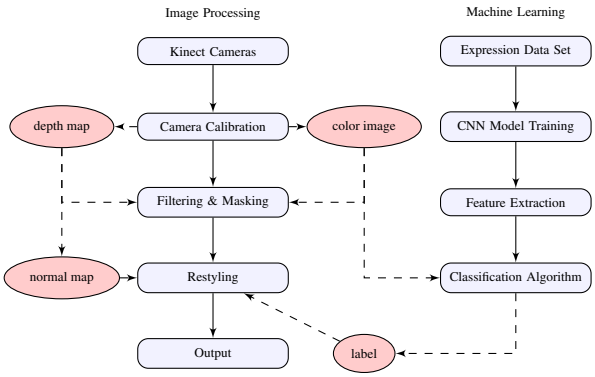



Figure 1: System architecture

## 3. Dataset and Features

The training dataset used for this project came from the second version of the Cohn-Kanade database called Cohn-Kanade Plus (CK+/CKP)[7]. CK+ includes 593 sequences of images across 123 subjects, and each of the sequences contains photos of a human face changing from neutral (no expression) gradually to peak expression. 327 sequences among the 593 have been given an expression label among 7 prototypical emotions: Anger, Contempt, Fear, Disgust, Happiness, Sadness and Surprise. We used the 327 sequences with label as our dataset, the total number of the images is 5876. We generated a large and a small dataset for training, the large dataset contains the first image in a person's expression sequence labeled neutral and the last four dramatic expression labeled according to the expression. The smaller one took only the first and the last two images.
The test dataset came from the original Cohn-Kanade dataset which is very similar to the mentioned CK+ dataset but without emotion labels. We manually labeled a total of 764 images from CK dataset with 8 categories similar to CK+.

## 4. Method

### 4.1. Overview

The project can be divided into two parts of image processing and machine learning. An overview of the project layout is depicted in Fig. 1.

### 4.2. Kinect

This project uses a Kinect for Windows® V2 module. Kinect consists of an infrared(IR) camera and a color (RGB) camera. Fig. 3a shows the outside and camera geometry for a Kinect model. The IR camera has a resolution of 512 $\times$ 424 pixels the RGB camera has a resolution of 1920 $\times$ 1080 pixels. The field of view is 70 $\times$ 60 degrees while frame rate rates at 30 frames per second with operative measuring range from 0.5 to 4.5 m.

### 4.3. Face tracking

The project perform face detection using a pre-trained Haar feature-based cascaded classifier to mark the region of interest(ROI) for the later image processing. The project uses Haar cascade classifiers for this task because they are simple, and most importantly, fast enough to achieve real-time classification on a modern laptop.

### 4.4. Expression Recognition

#### 4.4.1 Model Training

The neural network training platform Caffe[8] is used to achieve feature extraction from the image dataset. After CNN model training, we replace the built-in classification model with our own learning algorithm. The training result of raw Caffe data also give us an insight of the expected performance of our classification algorithm.

**Environmental Settings** Two major models were used in this project. The first model is inherited from [9](DeXpression) and the other is the well-known GoogLeNet model [10]. We inherited and fine-tuned the structure and the parameters from both models and used our dataset to retrieve the preliminary training result.

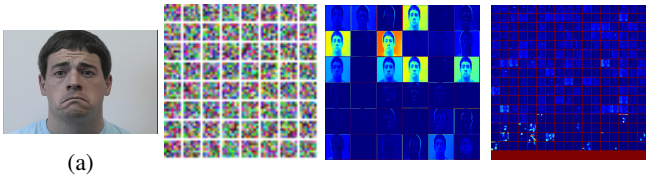
**Training results** We completed the of training the data set with established DeXpression and GoogLeNet model with 400000 and 240000 iterations. With the preliminary results in mind, we tuned the layer by layer parameters as well as determined the feature extraction interruption in the trained model and apply further algorithms to classify unseen data with higher precision.

**Facial Feature Extraction** After training with Caffe using our data set, we extracted parameters (filters) for each layer, thus extracting the feature information afterwards. Figure Fig. 2a is one example from the CK+ dataset of label sad. Figure Fig. 2b shows the filters used for convolution layer 1. Figure Fig. 2c Fig. 2d with their corresponding extract features (filter output). In both figure sets, the left is the original photo, the middle is the first 36 outputs of the first convolution layer, and the right are the final features, which are the output of the very last pooling layer. We will use these features to do further learning tasks.

### 4.5. Classification Algorithm

#### 4.5.1 Feature

After training the CNN model, we extracted the output of the last fully-connected layer for each input image as its features. Every image in the dataset is represented as such $1\times8$ vector of numbers. Part of the evaluation of algorithms was using MATLAB [12]



(a) (b) (c) (d)

Figure 2: Neural Net Layers. (a) Dataset photo. (b) First convolution layer visualization. (c) Feature extraction after first pooling. (d) Feature extraction after final layer

#### 4.5.2 Algorithms

After running training on DeXpression model, we would like to further enhance the performance of such so we evaluated different classification algorithm performances.

**Softmax** The original classifier of the neural network models were using softmax classification algorithm. Generalized from logistic regression binary classifier, softmax gave the likelihood of multi-class occurrences. For an image $x_i$ parameterized by weights $W$, the possibility of giving a correct label $y_i$ is calculated as

$$p(y_i|x_i, W) = \frac{e^{f_{y_i}}}{\sum_j e^{f_j}}$$

where $f$ is the functional mapping $f(x_i; W) = Wx_i$

**K-Nearest Neighbors(KNN)** Given a set of N points in an n-dimensional feature space, calculate proximity of the test point with the closest k neighbors using their relative Euclidean distances. Assign the test point to the class that has the most frequent occurrence.

### 4.6. Rendering

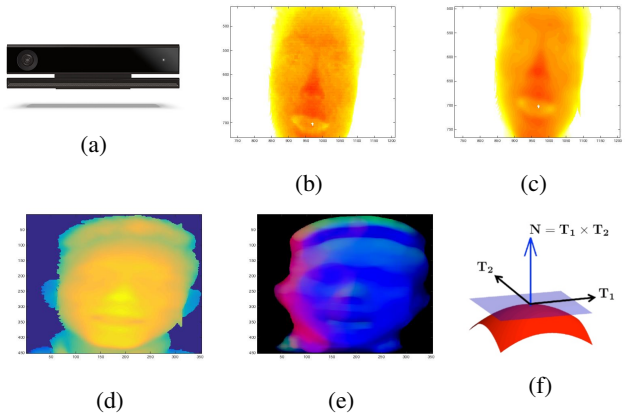
#### 4.6.1 Image Processing

Using morphological image processing as well as a bilateral filter, the depth map can be hole-filled and the noise were reduced while preserving edges. The second step of finer cropping the ROI is using the processed depth information to crop our ROI. Using a distance threshold of 2000(mm) combined with the previous face tracking coordinates, we are able to segment out the face in RGB and depth images.

#### 4.6.2 Restyle

In order to render more realistic lighting with the face, we calculated facial normal map using cross product of a small window's $x$ and $y$ direction vector as depicted in Fig. 3. Fig. 4.

Normal map of the face is used to determine the lighting of a particular pixel. A Lambertian reflectance model is used




(a) (b) (c) (d) (e) (f)

Figure 3: (a) A Kinect. (b) Unfiltered depth data. (c) Bilateral-filtered depth data. Normal map calculation. (d) Depth data. (e) Corresponding normal map(R:x, G:y, B:z). (f) Normal vector calculation

(a) (b) (c) (d)

Figure 4: Relighting. (a) Original plane photo. (b) Relit from -x direction with blue light. (c) Relit from +y direction with green light. (d) Relit in +z direction with white light.

to calculate the pixel color and intensity in relighting.

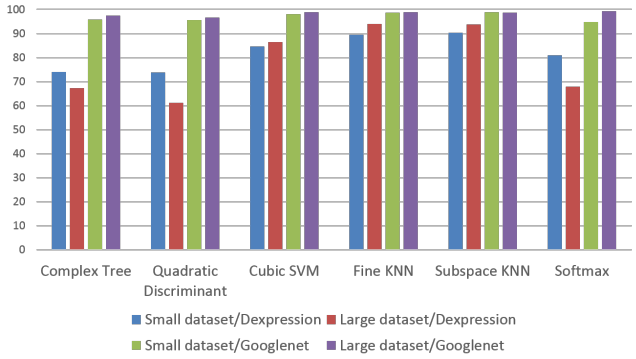
$$I_D = I_{org}\mathbf{L} \cdot \mathbf{N}CI_L = I_{org}|L||N|cos(\alpha)CI_L$$

where $I_{org}$ is the original pixel value, $I_D$ is the resulted pixel value, $L$ and $N$ are the light vector and normal vector, $C$ and $I_L$ are the color and intensity of the incoming light, and $\alpha$ is the angle between $L$ and $N$. Combining relighting as

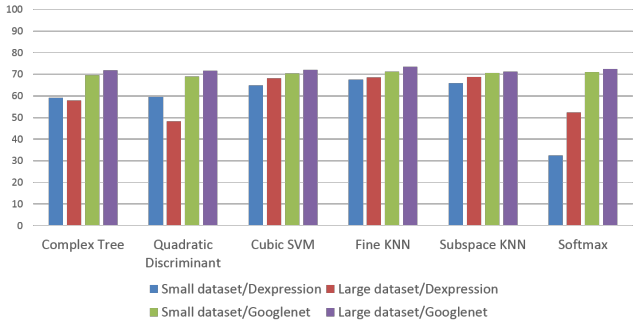
## 5. Results

### 5.1. CNN Training

We ran the GoogLenet model for 240000 iterations with an average loss of 0.005. The overall model accuracy is around 70% tested with 764 labeled data different from the training data. Expressions such as neutral, anger, happy, and sad were more precise than contempt, suprise, and fear. Learning rate was set to be 0.00001 different lower than the predefined 0.0001 since we were retraining the model with different dataset using the same weights. The dataset used was considerably small of only 1600+ images. Since the CK+ dataset also contains subtle expression of nearly neutral, those were not used since if labeled 'neutral', the number of data will largely overpower the others in quantity.




(a)




(b)

Figure 5: Testing accuracy (a)Cross validation. (b)Unseen Data

## 5.2. Accuracy

### 5.2.1 Accuracy Calculation

**Cross-Validation** As seen in Fig. 5a, six algorithms were evaluated using the same input from the last layer of features in the neural network. Using 5-fold cross-validation, the best accuracy achieved were softmax at 99%, and fine KNN and subspace KNN at 98.8%.

**Unseen Data** To further validate the system accuracy in real world application, we evaluated the accuracy with unseen dataset.f The best accuracy obtained was fine KNN with GoogLeNet on a larger dataset at 73.46%. We overfitted a little bit with the high accuracy of cross-validation and the lower for unseen data since this is the first time we implement such dataset with the models.

### 5.2.2 Model Comparison

As seen in both Fig. 5a and Fig. 5, the overall performance of DeXpression model is not as good as GoogLeNet with different classification algorithms. The two models only had similar performances during the test phase using cubic SVM and KNN algorithms. On reason for the performance differences may be the number of layers for GoogLeNet is nearly three times the size of DeXpression, which allows it to pick up finer features.

### 5.2.3 Algorithm Comparison

The best performed algorithm in the cross-validation phase is softmax with GoogLeNet model on larger dataset. The best performed algorithm in the test phase was fine KNN with GoogLeNet on a larger dataset. From the graphs, we can see the performance of softmax heavily depends on CNN model and dataset, whereas KNN and SVM were more robust toward model and data variation.

### 5.2.4 Dataset Comparison

The variation of dataset(large or small) had a significant effect on the performance of softmax classification as seen in both graphs. SVM and KNN did not have large variations especially for GoogLeNet in the testing phase.

## 5.3. Misclassification Analysis

When trying to do emotion classification, we found out that there are several classes that are frequently misclassified. In the training phase, the best accuracy is 99.3%, obtained using softmax algorithm with feature extracted using GoogLeNet CNN model. Under this setting, 5% of 'neutral' faces are misclassified to be 'contempt'. As shown in the confusion matrix (Fig. 6e). This weak point of the classification may result from the subtle difference between facial expression 'contempt' (Fig. 6a) and 'neutral' (Fig. 6b).

In the testing phase, the best accuracy is 72.9%, obtained using subspace KNN with feature extracted using GoogLeNet CNN model. (Fig. 6f) is the confusion matrix of the prediction results under this setting. 29% of 'angry' faces are misclassified to be 'contempt'. This tendency may again be caused by the similarity between facial expression 'angry' (Fig. 6a) and 'contempt' (Fig. 6b). Another worthmentioning result is that in the testing phase, none of the contempt images are predicted correctly (0% accuracy). This may be result from the skew of the training dataset, since the contempt-labeled images account for only 4.4% of the larger dataset. The tendency of higher misclassification rate on class having less examples can also be detected from the prediction result of 'sadness'. (30.7% accuracy, occupying 6.9% of the examples.)

## 5.4. Rendered Images

The rendered images can be seen in Fig. 7, picked from correct labeled events.

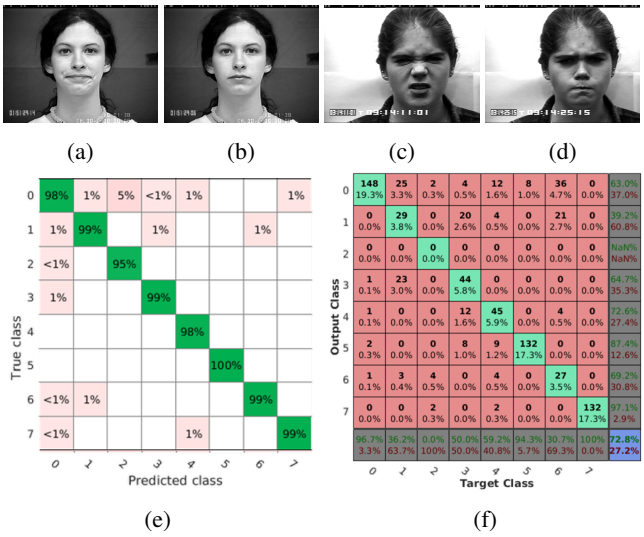



Figure 6: Classes prone to misclassification. (a) (b)Contempt versus neutral. (c) (d)Disgust versus angry. (e)Confusion matrix of softmax algorithm on GoogLeNet CNN features in train phase (cross validation). (f)Confusion matrix of subspace KNN algorithm using GoogLeNet CNN features on unseen data.The corresponding emotions to the labels 0 to 7 are: 0-neutral, 1-angry, 2-contempt, 3-Disgust, 4-happy, 5-fear, 6-sadness, 7-surprise


Figure 7: Final example images of (a) neutral (b) angry (c) contempt (d) disgust (e) fear (f) happy (g) sadness (h) surprise

# 6. Conclusion and Future Work

## 6.1. Future Work

The system currently is dealing with only front-facing portrait mode, but did not test its capability in side-way facial images. If newer techniques can compensate the incomplete face recognition or deploy a multi-camera system, a real-time tracking and editing system can be built.

This project, considered as a prototype, proved the feasibility of automatic visual effects, but is not yet optimized for real-time application since neural network evaluation takes around 1 second per frame. If algorithm and hardware are optimized and neural network calculation can be done in GPUs, the overall algorithm will be significantly more time efficient.

This project modifies existing neural network models for expression recognition, but a customized neural network model can be proposed for this task in order to optimize the performance.

Limited by the resources, we were only able to use CK dataset of up to around 1600 images which, for an eight category classification task, is too small. Larger datasets or combination of datasets can be used in further research to deliver better results.

## 6.2. Conclusion

The report have demonstrated the capabilities of AI in automatic rerendering. Combining machine learning algorithms with existing image processing tools, we presented an integrated system which takes in 3D human portrait to perform restyling base on the emotion of the character.

The automatic restyling system consists of three components, the hardware interface for image acquisition, machine learning for expression recognition, and image processing for rerendering and editing. For image acquisition, the Kinect camera modules work well producing encouraging results. For expression recognition, using CNN is proven to be a promising approach as shown in the previous work. To achieve higher accuracy, we integrated CNN with KNN, with CNN responsible for feature generating and KNN performing classification. By replacing the softmax layer in the neural network with KNN classifier, we got a 16.5% boost in accuracy when using DeXpression CNN model, and a 1% boost in accuracy when using GoogLeNet CNN model. Also, by comparing multiple classifiers, we found out that compared with softmax, KNN is less sensitive to the quality variation of the input feature. In another words, KNN has a smaller performance difference between the results using features extracted from neural networks with different complexity. As for image processing and rendering, well known filters and morpholigical image processing methods provide efficient and effective tools to fulfill the needs. Although the depth information from Kinect is not as good in resolution as the color images, it still provides sufficient structural information to be used to relight the face.

With the growing market of live streaming and social media platforms with quick and easy editing, there is no doubt that incorporating artificial intelligent special effect editors will be an useful tool in the market someday.

## References


[1] Matthew Turk and Alex Pentland. *Eigenfaces for Recognition* Journal of Cognitive Neuroscience, IEEE(1991): Vol. 3, No. 1, Pages 71-86

[2] M. Lyons, S. Akamatsu, and M. Kamachi. *Coding facial expressions with Gabor wavelets.* Automatic Face and Gesture Recognition, IEEE(1998). Proceedings.

[3] Fasel, B. *Robust Face Analysis Using Convolutional Neural Networks.* Object Recognition Supported by User Interaction for Service Robots (2002): 1-48.

[4] Yu, Zhiding, and Cha Zhang. *Image Based Static Facial Expression Recognition with Multiple Deep Network Learning - Microsoft Research.* Microsoft Research. IEEE(2015).

[5] Matthias Ziegler, Andreas Engelhardt, Stefan Mller, Joachim Keinert, Frederik Zilly, Siegfried Foessel. *Multi-camera system for depth based visual effects and compositing* CVMP(2015).

[6] Yang Wang, Lei Zhang, Zicheng Liu et.al *Face Relighting from a Single Image under Arbitrary Unknown Lighting Conditions.* Transactions on Pattern Analysis and Machine Intelligence, IEEE(2009):Vol. 31, Issue: 11.

[7] Lucey, Patrick, 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.* Computer Society Conference on Computer Vision and Pattern Recognition Workshops ,IEEE(2010): n. pag. Web.

[8] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell. *Caffe: Convolutional Architecture for Fast Feature Embedding.* 22nd ACM international conference on Multimedia, ACM(2014): 675-678.

[9] Peter Burkert, Felix Trier et al. *DeXpression: Deep Convolutional Neural Network for Expression Recognition*. German Research Center for Artificial Intelligence, DFKI(2016).

[10] Szegedy, Christian, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. *Going Deeper with Convolutions.*. Conference on Computer Vision and Pattern Recognition, IEEE CVPR(2015): Web.

[11] Vinyals, Oriol, Blundell, Charles, Lillicrap, Timothy, Kavukcuoglu, Koray, and Wierstra, Daan. *Matching networks for one shot learning*. arXiv preprint arXiv:1606.04080, 2016.

[12] MATLAB Statistics and Machine Learning Toolbox Release 2016a, The MathWorks, Inc., Natick, Massachusetts, United States.

[13] Lingzhu Xiang, Florian Echtler, Christian Kerl, Thiemo Wiedemeyer, Lars, hanyazou, Alistair . (2016). libfreenect2: Release 0.2 [Data set]. Zenodo. http://doi.org/10.5281/zenodo.50641

[14] Zhen Wen, Zicheng Liu, Thomas S. Huang. *Face Relighting with Radiance Environment Maps.* CVPR(2003).

[15] Lembit Valgma. *3D Reconstruction Using Kinect v2 Camera.* Bachelor's thesis, 12 ECTP(2016)

[16] Jan Smisek, Michal Jancosek, and Tomas Pajdla. *3D with Kinect.* Chapter 1, Consumer Depth Cameras for Computer Vision, ACVPR(2016).