Towards Improved Facial Key Point Detection With Deep Convolutional Neural Networks

Filemon Mateus
u1419667@utah.edu

Wei Zeng
u1330357@utah.edu

1 Keywords

— Convolutional Neural Network (CNN)
— Fully Connected Neural Network (FCNN)
— Residual Neural Network (ResNet)
— Masked Loss Residual Neural Network (ML-ResNet)

2 Repository

— Hosted and publicly accessible at: https://github.com/WayGold/FacialKeypointDetection.

3 Introduction

The advent of technological advances in the past few decades has ignited the video game industry to witness an unprecedented amount of progress in design, graphics, and animation. This progress has been one of the major leading forces in the production of extremely immersive AAA titles that make the video game industry a multi-billion dollar global enterprise. However, due to the rapid transformative nature of these technologies, practitioners, developers, and animators in this space have become increasingly burdened with the inherit specificities of an always changing technology. One canonical example here is the performance capturing technology, or more specifically, the facial capturing technology.

In the past, prior to the introduction of facial performance capturing technology, traditional face animation tasks relied heavily on the animator manually selecting keyframes from a clip to create an immersive result. However, with the introduction of facial capturing technology, this task has been reduced to creating a generic profile for each actor and make subsequent adjustments (if necessary!) across different clips. In softwares like Dynamixyz [1], animators usually necessitate 86 key points—each with 25 selected frames from a ROM video[2]—to produce an acceptable result. After the selection of these points, a tracking profile is generated and used recurrently across the clip to track facial expressions. However, this only constitutes the very first step before retargeting and rendering the clip to the game model, so it is already evident how cumbersome facial point detection tasks can be, especially for AAA titles that exhibit many cutscenes with different actors in them.

4 Methods and Experiments

Similar to the seminal methods employed in [1] and [2], we propose Deep Convolutional Neural Networks as a potential solution to mitigate the aforementioned burden and help speed up facial animation/capturing

[1] Dynamixyz is the state-of-the-art software for facial motion capture. It is renowned as a world-class leader in video-based facial animation software.
[2] A ROM video isn't the same as a performance clip, instead it refers to a clip that contain a set of extreme emotions performed by the actor to calibrate its corresponding generic profile during an animation pipeline.

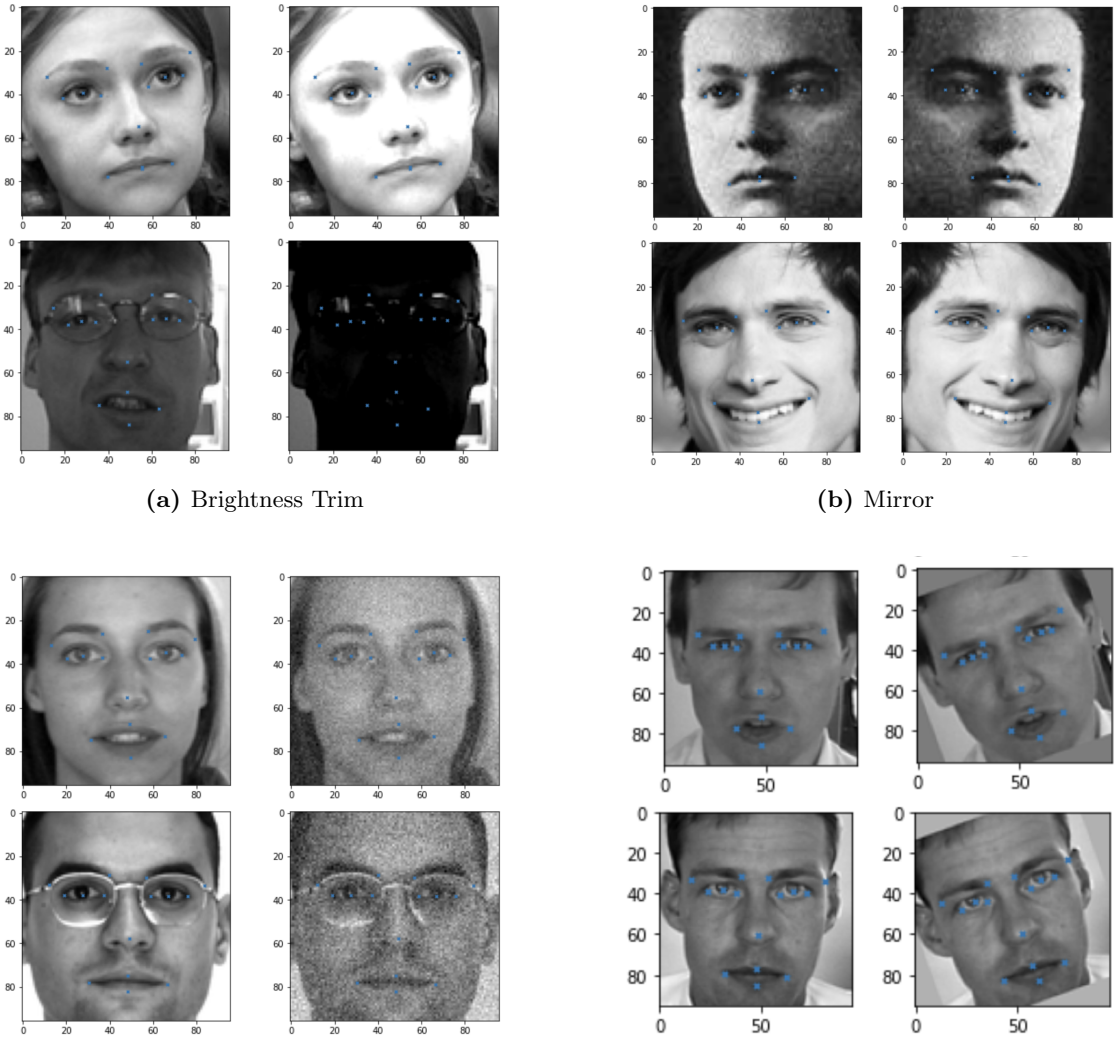
production pipelines, and achieve competitive results in scenarios with missing and limited annotated labels.

Facial Keypoint Detection is an extremely difficult topic, and although computer vision research has made significant strides in tackling some of the most pressing problems in it, there is still much room for improvement. As seen in [1] and [3], it is evident that these methods are producing reliable results, but additional research is necessary to improve component detection and robustness against variations in pose and expression in scenarios where data is noisy, incomplete, or missing annotated labels.

We propose the utilization of different CNN architectures with varying choices of hyperparameters to aid in the automatic detection and localization of keypoints on face images. We test architectures such as ResNet and ML-ResNet [1] against a simple baseline benchmark in the form of a Fully Connected Neural Network to measure the efficacy of Deep Neural Networks in Facial Keypoint Detection tasks.

5 Dataset

Our dataset comes from the Kaggle Facial Keypoint Dataset [4] and was graciously provided by Dr. Yoshua Bengio of the University of Montreal. The training set contains 7049 images and 15 facial key points representing different facial features like the brows, eyes, nose, and lips (but with some missing annotated points). The target variables are represented by these facial key points. There are 1783 images total in the test dataset, so collectively, we have a dataset of 8,879 images consist of 96×96 pixel images with a single channel dimension (greyscale images). To combat missing labels, we perform data augmentation and the image processing methods introduced in Longpre *et al* [2] (see exemplars in Figure 1).



(a) Brightness Trim

(b) Mirror

(c) Noise

(d) Rotation

Figure 1. Augmented Facial Keypoint Dataset
https://www.kaggle.com/competitions/facial-keypoints-detection/overview.

6 Network Architectures

6.1 Fully Connected Network

Our first architectural choice is a Fully Connected Neural Network that takes as input a $96 \times 96 \times 1$ greyscale image vector $\mathbf{x}$ and outputs a 30–dimensional vector $\mathbf{y}$—corresponding to the x and y coordinates of each of the 15 distinct facial features as seen in Figure 1. Structurally, there is only one *hidden layer* augmented with a ReLU activation function that applies non-linearities to the input $\mathbf{x}$. We use a Fully Connected Neural Network as baseline architectural choice for experiments in section 8. A visual representation of the structure and its constituents parts is provided below.

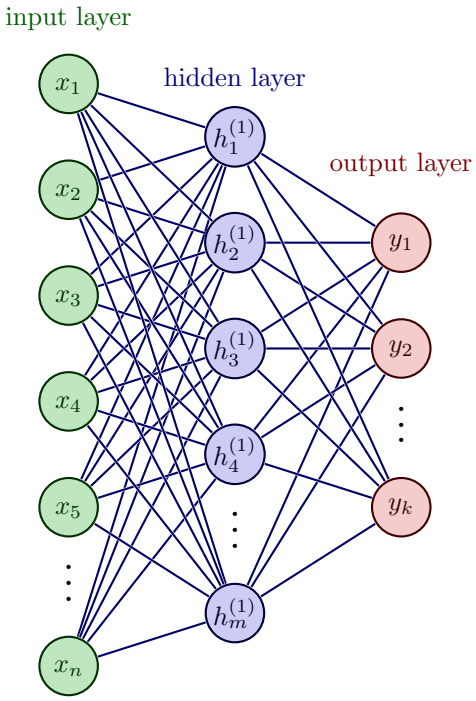


Figure 2. Fully Connected Neural Network. For our experimental purposes in section 7 we have fixed choices of $n = 9216$, $m = 500$, and $k = 30$.

6.2 Deep Residual Neural Network

Inspired by [1], our second architectural choice rests in a Deep Residual Neural Network, whose constituent parts is illustrated in Figure 3. Essentially, this is a residual neural network whose residual blocks are made up of two convolutional layers with a shortcut connection from the input to the output of the second batch normalization layer.

Although our ResNet could potentially contain any number of residual blocks, due to the small size of the training data, we opted for relatively smaller number of residual blocks to prevent overfitting and make the ResNet feasibly trainable on portable machines.

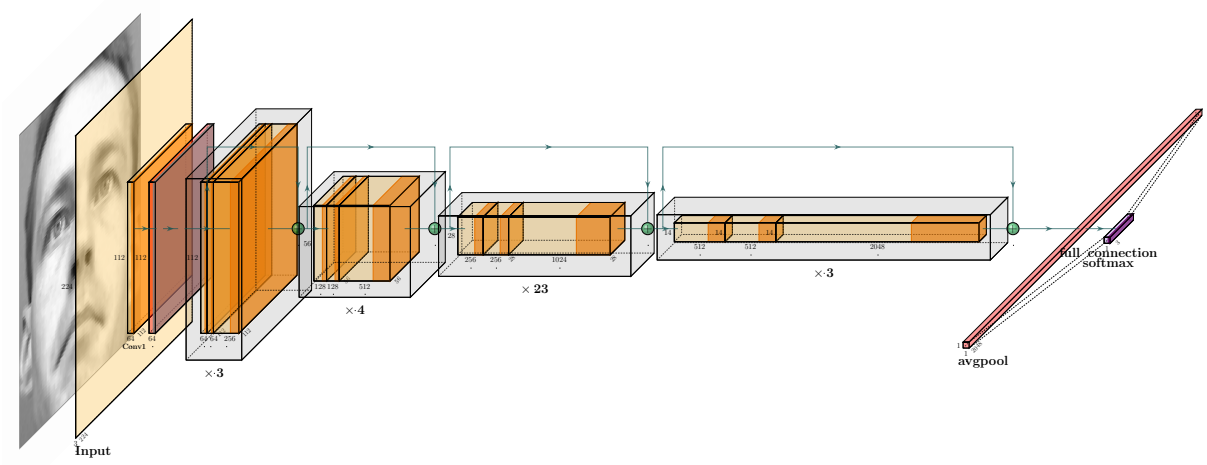
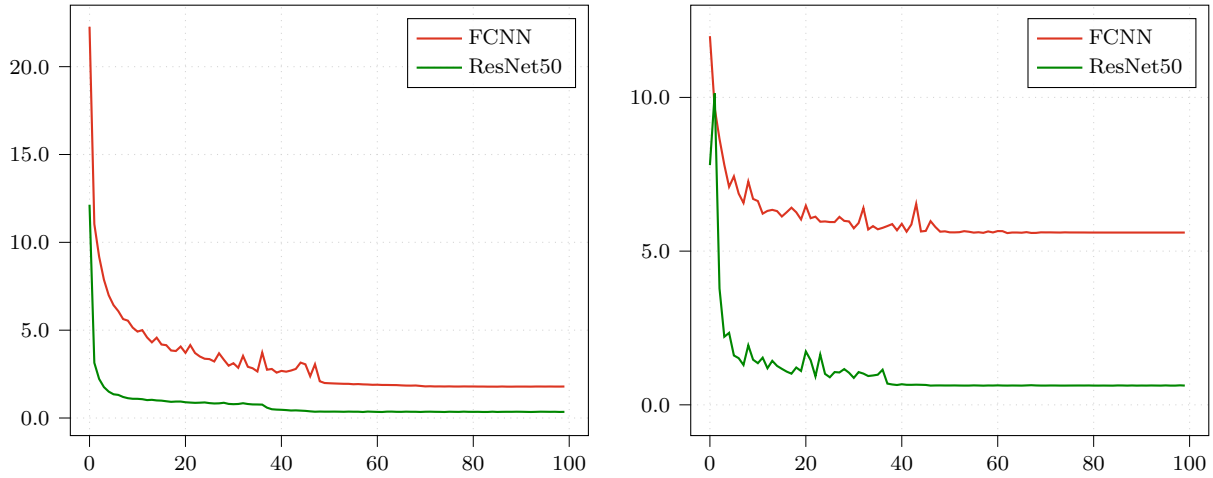


Figure 3. Deep Residual Neural Network — ResNet50.

7 Results

Our results suggestively demonstrate a decisive superiority in Facial Keypoint Detection by the ResNet50 model which outperformed the Fully Connected model by large percentual points as seen in Table 1. Below, it is an illustration showing the relative faster training convergence of the ResNet50 model against its baseline Fully Connected Neural Network.

7.1 Training and Validation



(a) Training Loss FCNN vs ResNet50.

(b) Validation Loss FCNN vs ResNet50.

Figure 4. Historical performance on the Facial Keypoint Dataset over 100 training epochs.

# Epochs	Training Loss		Validation Loss	
	FCNN	ResNet50	FCNN	ResNet50
0	22.27	12.14	11.99	7.80
10	6.63	1.09	6.63	1.36
20	3.71	0.90	6.48	1.74
30	3.12	0.79	5.74	0.88
40	2.68	0.47	5.89	0.67
50	1.98	0.36	5.61	0.63
60	1.90	0.35	5.65	0.63
70	1.80	0.36	5.61	0.63
80	1.79	0.35	5.60	0.63
90	1.79	0.36	5.60	0.63
100	1.79	0.35	5.60	0.62

Table 1. Training/Validation convergence FCNN vs ResNet50.

7.2 Predictive/Inferencial Results

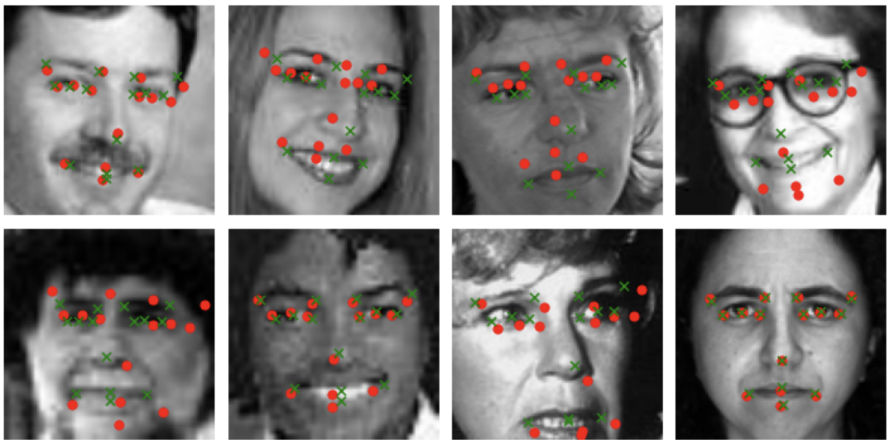


Figure 5. Comparative inferential results on the test set. Fully Connected Neural Network is outline in red while the ResNet50 is depicted in green.

Figure 6. Predictive results on a subset of annotated valid keypoints. Fully Connected Neural Network is outline in red while the ground-truth label is depicted in green.

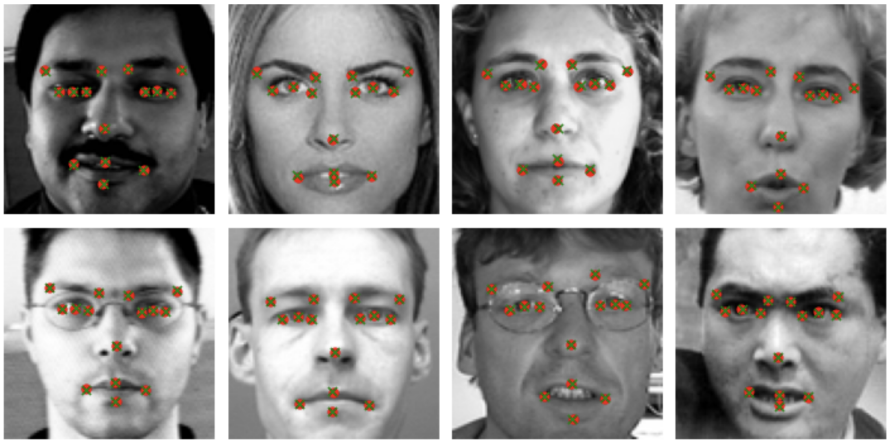


Figure 7. Predictive results on a subset of annotated valid keypoints. ResNet50 is outline in red while the ground-truth label is depicted in green.

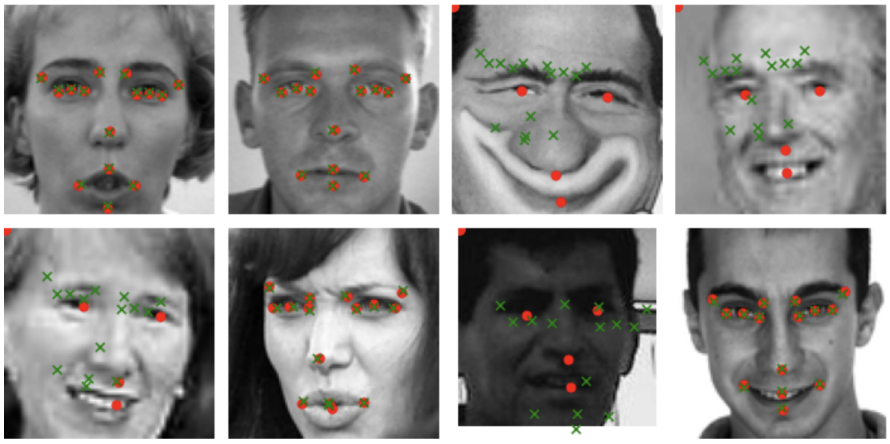


Figure 8. Inferential results on noisy dataset with autofilled, augmented data entries. Fully Connected Neural Networkis outline in red while the ground-truth label is depicted in green.

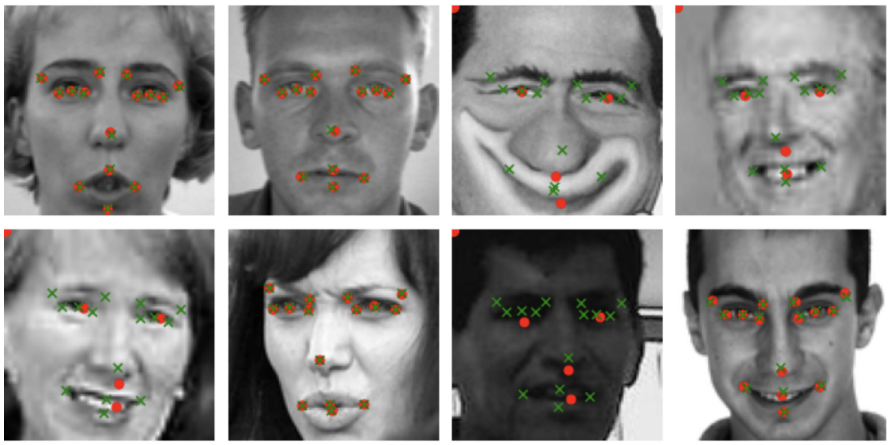


Figure 9. Inferential results on noisy dataset with autofilled, augmented data entries. Resnet50 is outline in red while the ground-truth label is depicted in green.

8 Conclusion

In this work, we successfully conducted experiments on the Facial Keypoint Detection dataset and leverage the power of Deep Residual Neural Networks to improve generalizability across the detection and localization of keypoints on face images on instances where data is noisy, incomplete, or missing annotated labels. Further developments on the course of this project includes the utilization of the masked loss function to improve robustness during training, and reduce inferential inconsistencies during prediction on noisy data entries with missing annotated labels.

9 References

[1] S. Wu, J. Xu, S. Zhu, and H. Guo, "A deep residual convolutional neural network for facial keypoint detection with missing labels," *Signal Processing*, vol. 144, Nov. 2017. DOI: 10.1016/j.sigpro.2017.11.003.

[2] S. Longpre and A. Sohmshetty, "Facial keypoint detection," 2016. [Online]. Available: http://cs231n.stanford.edu/reports/2016/pdfs/010_Report.pdf.

[3] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar, "Localizing parts of faces using a consensus of exemplars," in *CVPR 2011*, 2011, pp. 545–552. DOI: 10.1109/CVPR.2011.5995602.

[4] "Facial keypoint detection kaggle dataset," 2022. [Online]. Available: https://www.kaggle.com/competitions/facial-keypoints-detection.