Head Pose Estimation from Images

Sanjeet Swaroop Panda [1] **Souvadra Hati** [2]

Abstract

Being able to estimate the pose of a person in front of us informs us lot more information regarding the intentions of that person and other nonverbal cues. We, human beings can do that analysis very easily from an early age (1). In the field of Computer Vision research, although detecting the face of a person is now an almost solved problem, modern day Computer Vision algorithms still suffer a lot to accurately determine head poses (characterized by *pitch, roll, and yaw* (1). In this project we have tried to come up with deep-learning based approaches (particularly a solution, that can be implemented with ease) to solve this age-old problem and reach a decent level of accuracy in that task.

Technical Details

This problem has been tackled by many scientists via different methods (2) (**?**) (3), but deep-learning and specially Convolutional Neural Network (CNN) based approaches have been implemented only recently (4; 5). This was one of the motivations for us to try to find a CNN based solution to this problem.

A transformer is a deep learning model that adopts the mechanism of attention, and weighing the influence of different parts of the input data. This popular Natural Language Processing algorithm is increasingly being used in Computer Vision problems as well. It works based on attention and auto-encoder.

$$Attention(Q, K, V) = softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

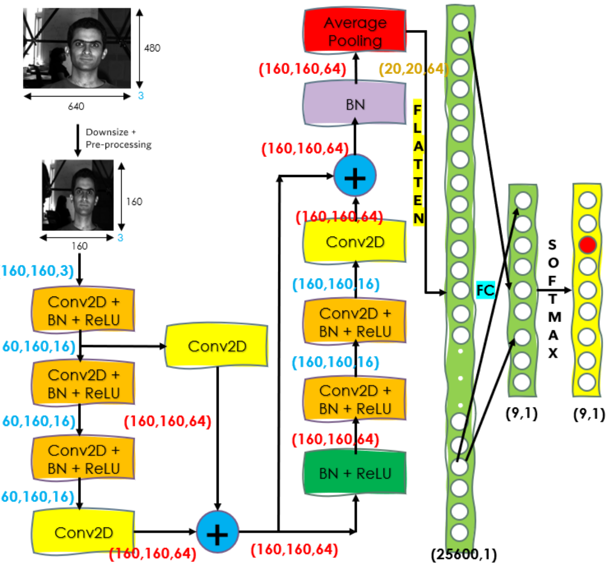
[Q: Query matrix, K: Key Matrix, V: value matrix, d_k: dimension of key matrix]

Our assigned data was "The Extended Yale Face Database B" (6), which is 16128 images of 28 human subjects under different light conditions labelled w.r.t their poses, azimuthal and elevation angles. One particular challenge present in this dataset is that some of the light conditions make the images almost completely dark (unlike most other head pose estimation databases (4; 5)) that makes the learning process significantly difficult. The labelling of the given data provided us with two ways to model the problem:

- Use the **pose** labels to define a classification model.
- Use the **azimuthal** and **elevation** angles to define a regression model.

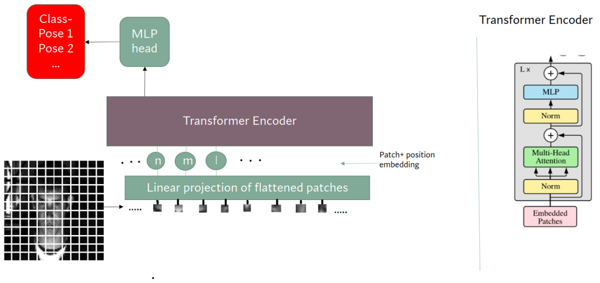
Since, most of the literature on CNN were on classification problems, it was a natural choice to go with the first option (along with the fact that our regression approaches did not work out well).

We have proposed a CNN based architecture inspired by ResNet-34 (7) and IRHP-Net (5) architectures, with keeping in mind the goal of making a network that can be easily built using 'off-the-shelf' Tensorflow (8) building blocks. Our custom neural network looks like:



We also attempted an innovative transformer based model inspired by the success of the recent vision transformer in various fields of computer vision such as segmentation classification etc. We used a vanilla vision transformer inspired by Dosovitskiy et al. (2020) (9)

[1] UG 3rd year, BS in Physics, Indian Institute of Science, Bangalore [2] UG 3rd year, BS in Biology, Indian Institute of Science, Bangalore. Correspondence to: Sanjeet <sanjeetpanda@iisc.ac.in>, Souvadra <souvadrahati@iisc.ac.in>.



Results

We have trained our CNN based model numerous times on the data-set and found that its average accuracy is $\approx 72\%$, where some initial conditions have provided accuracy of 76.3%. But, this accuracy is not as high as mentioned by models described in (4; 5) due to the fact that most of the false positives were due to poor light conditions hiding important features of the face, such as eyes and sometimes the whole face.

While our transformer model didn't perform as well as we expected it to, after hyper-parameter tuning and access to more computational resources we expect to get higher performance.This is because, prior results (9) show that attention based models in general perform better than vanilla CNN based models.

Novel Contributions

After extracting the features learnt by the CNN model, we can visualize that it is learning the right features of the head to estimate its pose (e.g. eyes, necklines, head edges ...). Moreover, on our data-set, our model is one of the best model in overall classification of the data with accuracy 76%, that surpasses the previous attempt (10) on the full data with accuracy of 62%.

Tools used

We have implemented our CNN based model in Python 3.8 using the following open-source tools:

- Numpy (11)
- Tensorflow (8)
- OpenCV (12)

Individual Contributions

- CNN based literature review: Souvadra Hati
- Non-CNN literature review: Sanjeet Swaroop Panda
- Data acquisition code: Souvadra Hati
- CNN-based model and experiments: Souvadra Hati
- Vision Transformer model and experiments: Sanjeet Swaroop Panda
- Report and Presentation: Both

References

[1] E. Murphy-Chutorian and M. Trivedi, "Head pose estimation in computer vision: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 31, pp. 607–26, 05 2009.

[2] N. Kwak, S.-I. Choi, and C.-H. Choi, "Feature extraction for regression problems and an example application for pose estimation of a face," in *Image Analysis and Recognition* (A. Campilho and M. Kamel, eds.), (Berlin, Heidelberg), pp. 435–444, Springer Berlin Heidelberg, 2008.

[3] J.-M. Kim and H.-S. Yang, "A study on object recognition technology using pca in the variable illumination," in *Advanced Data Mining and Applications* (X. Li, O. R. Zaïane, and Z. Li, eds.), (Berlin, Heidelberg), pp. 911–918, Springer Berlin Heidelberg, 2006.

[4] M. Patacchiola and A. Cangelosi, "Head pose estimation in the wild using convolutional neural networks and adaptive gradient methods," *Pattern Recognition*, vol. 71, pp. 132–143, 2017.

[5] H. Liu, X. Wang, W. Zhang, Z. Zhang, and Y.-F. Li, "Infrared head pose estimation with multi-scales feature fusion on the irhp database for human attention recognition," *Neurocomputing*, vol. 411, pp. 510–520, 2020.

[6] A. Georghiades, P. Belhumeur, and D. Kriegman, "From few to many: Illumination cone models for face recognition under variable lighting and pose," *IEEE Trans. Pattern Anal. Mach. Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[7] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, 2016.

[8] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin, S. Ghemawat, I. Goodfellow, A. Harp, G. Irving, M. Isard, Y. Jia, R. Jozefowicz, L. Kaiser, M. Kudlur, J. Levenberg, D. Mané, R. Monga, S. Moore, D. Murray, C. Olah, M. Schuster, J. Shlens, B. Steiner,

I. Sutskever, K. Talwar, P. Tucker, V. Vanhoucke, V. Vasudevan, F. Viégas, O. Vinyals, P. Warden, M. Wattenberg, M. Wicke, Y. Yu, and X. Zheng, “TensorFlow: Large-scale machine learning on heterogeneous systems,” 2015. Software available from tensorflow.org.

[9] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, “An image is worth 16x16 words: Transformers for image recognition at scale,” *CoRR*, vol. abs/2010.11929, 2020.

[10] H. Oost, “Sda-based discrete head pose estimation,” November 2009.

[11] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, “Array programming with NumPy,” *Nature*, vol. 585, pp. 357–362, Sept. 2020.

[12] G. Bradski, “The OpenCV Library,” *Dr. Dobb’s Journal of Software Tools*, 2000.