Human Pose Estimation Using Deep Convolutional Neural Networks

Yogesh B
EE12B066

CS6350: Artificial Neural Networks
Profesor: Dr. Anurag Mittal
14 December 2015

Abstract

Human Pose Estimation is a classic problem in Computer Vision where we are required to detect the location of each of the body joints. In this project, we tackle this problem using a part detector and a Markov Random Field, both modelled as a deep Convolutional Neural Network. We experiment on various network architectures and compare their performance to the state-of-the-art techniques.

---

1 Introduction

Despite a long history of prior work, Human Pose Estimation remains a very challenging task in Computer Vision. Deformable Part Models were one of the most successful techniques for this task, since human body naturally segments into articulated parts. Traditional approaches have used hand-crafted features like HOG or SIFT followed by a standard classifier like SVM. Recently, Deep Learning techniques are very successful in learning robust feature representations for a given task.

In this project, following the work done by [1], we use Convolutional Neural Networks for Pose Estimation. The general framework involves a part detector, followed by a MRF-based spatial model. Part detector detects the probability distribution for the location of each of the body parts, whereas the spatial model accounts for the spatial constraints between the different body parts. Both the part detector and the Markov Random Field are modelled as Convolutional Neural Networks and the joint training of both the networks reach the state-of-the-art performance for this task.

We first implement the model proposed by [1], which we call as baseline approach. Then, for the rest of the project, we focus on obtaining improvements in the Part Detector. Inspired by many of the recent works in Deep Learning, we try out different architectures for the Part Detector and compare the performance to the baseline approach. We finally come up with a architecture that achieves marginal improvements over the baseline technique

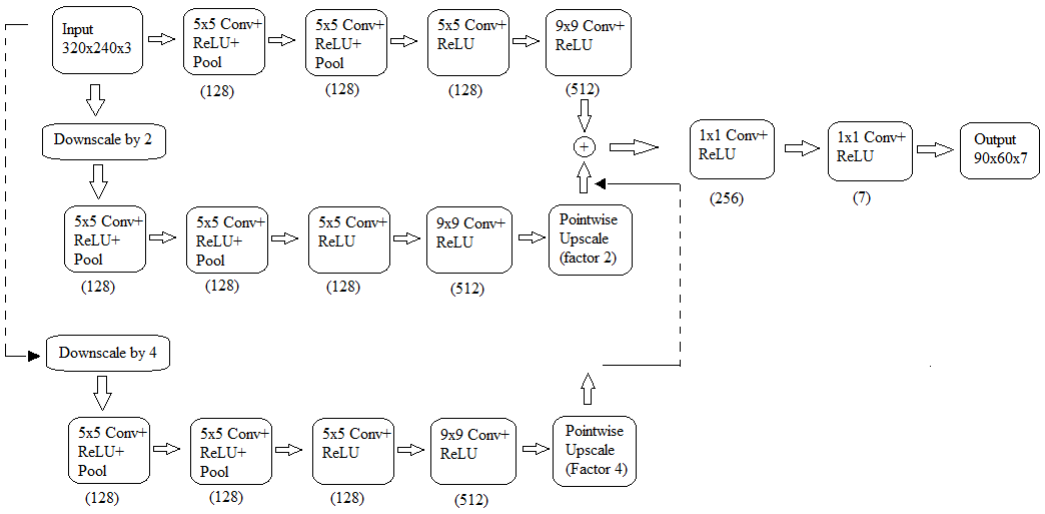
2 Baseline Approach

The first stage in the pipeline is the part detector, where the input is an RGB image consisting of people and the outputs are the heat-maps corresponding to each of the following parts - head, shoulder, elbow and wrist. These heat-maps are then passed to the spatial model which checks for consistency between different body parts and returns the refined heat-maps as output.

2.1 Part Detector

The work done by [1] proposes a sliding window detector, (i.e) a Comvolutional Neural Network which acts on a small region slid over the entire image. But this process is computationally expensive as there are a lot of redundant convolutions. So, they come up with an efficient equivalent model and use it in their final network. Moreover, they also show that using a multi-scale detector give best performance compared to a single detector. The approximate model for the part detector with three resolution banks is shown in Figure 1.

Figure 1. Part detector with three resolution banks



2.2 Spatial Model

The next stage in the pipeline is the spatial model, which is an MRF-like model over the distribution of spatial locations of each body part. The spatial model proposed by [1] connects each body part to every other part in a pairwise fashion to obtain a fully connected graph. The part detector provides the unary potentials for each body part, whereas the pairwise potentials are learnt in the convolutional model. Let us denote the unary potential for the body part A as P_A. The pairwise potential for the part A given B, denoted by $P_{A|B}(i, j)$ is the likelihood that part A occurs in location (i, j) given part B occurs in the center. With this notation, we can compute the final marginal likelihood as follows:

$$\bar{P_A} = \frac{1}{Z} \prod_{v \in V} (P_{A|v} * P_v + b_{v \to A}) \tag{1}$$

where $b_{v \to A}$ is the bias term used to describe the background probability and Z is the partition function. The above equation can be implemented in a Convolutional Neural Network, where the pairwise maps $P_{A|v}$ can be treated as the filters to be learnt. If there are n parts, then we need to learn n^2 filters. The input to the spatial model are the heat-maps from the Part Detector and the outputs are the ground truth heat maps. For numerical stability, we transform (1) to log domain as follows

$$\bar{e_A} = exp(\sum_{v \in V} [log(SoftPlus(e_{A|v}) * ReLU(e_v) + SoftPlus(b_{v \to A}))]) \tag{2}$$

where $SoftPlus(x) = \frac{1}{\beta} log(1 + exp(\beta x))$

The network architecture of the spatial model is shown in Figure 2.

Figure 2. Spatial Model

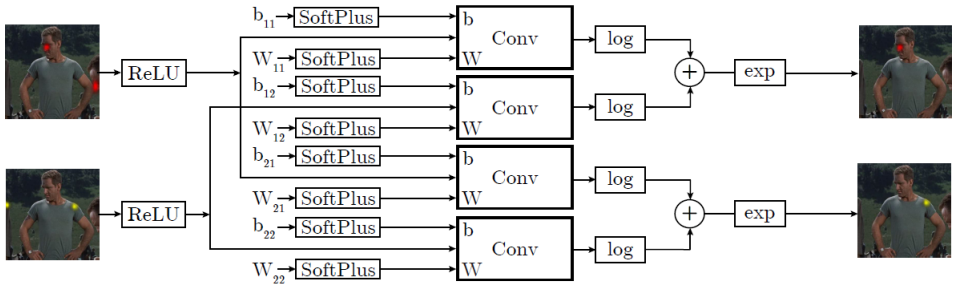


Figure 6: Single Round Message Passing Network

3 Experiments

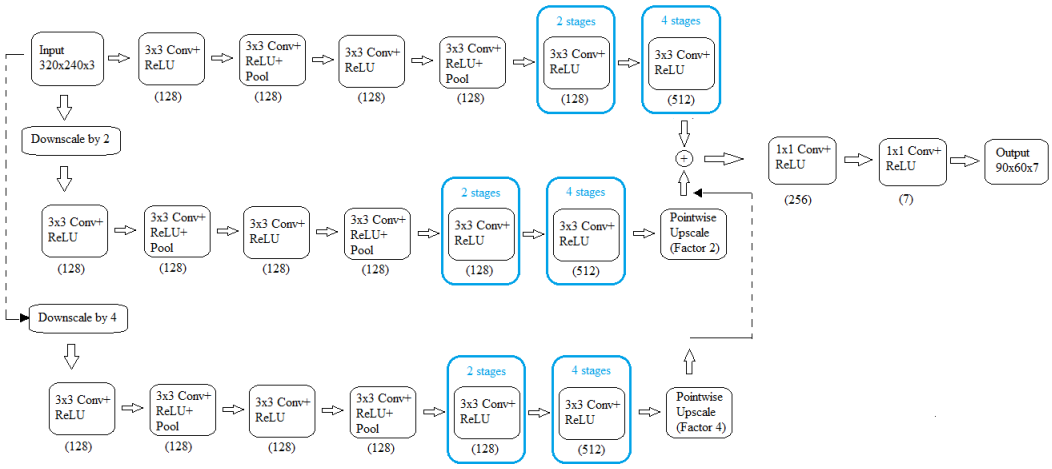
In this project, we propose three Convolutional Network architectures for the part detector phase and compare their performance to the baseline approach. The baseline approach consist of three resolution banks - the actual image gets passed to the first bank, whereas, the second bank sees the image downsampled by a factor of 2 and the third bank sees the image downsampled by a factor of 4. In all our proposed approaches, we use three resolution banks.

3.1 Network 1

Recently, a lot of works ([2],[3]) have shown improved performance in Image recognition task by replacing large Convolution filters with multiple smaller filters, typically $3X3$ filters. We extend the same idea to the part detector of the baseline model. But, there is no way to decide how deep our network should be. So, in the Network 1, we replace the larger filters with n $3X3$ filters, such that the region spanned by the larger filter is equivalent to the span

of n $3X3$ filters. For example, a $5X5$ filter is replaced by 2 $3X3$ filters. The architecture of Network 1 is shown in Figure 3.

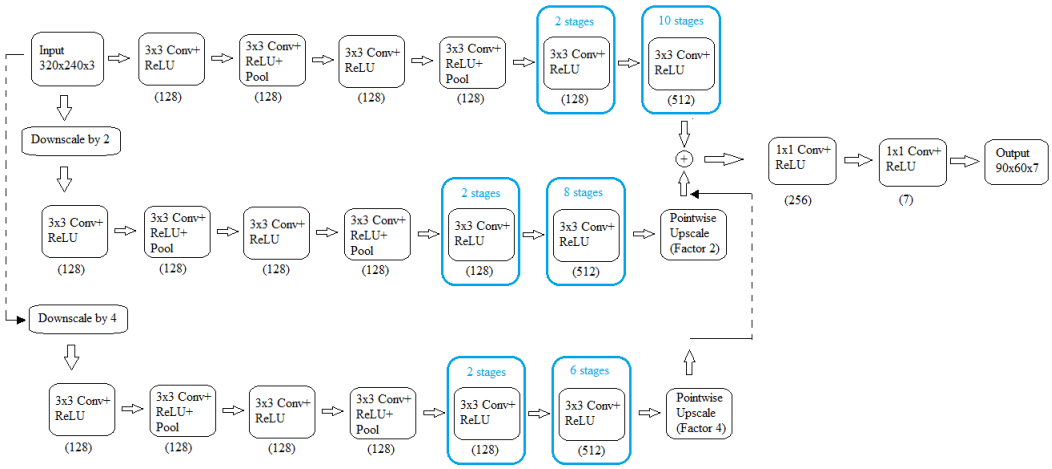
Figure 3. Architecture of Network 1



3.2 Network 2

Network 2 is a modification of Network 1, where the depth of the network is increased. Instead of increasing the depth equally across all the resolution banks, we perform a variable depth increase, where the highest resolution bank gets a deeper network compared to the lower resolution banks. The intuition behind this is that since higher resolution banks see finer details in the image, they require better features as compared to lower resolution banks, whose main purpose is to capture the context information. The architecture of Network 2 is shown in Figure 4.

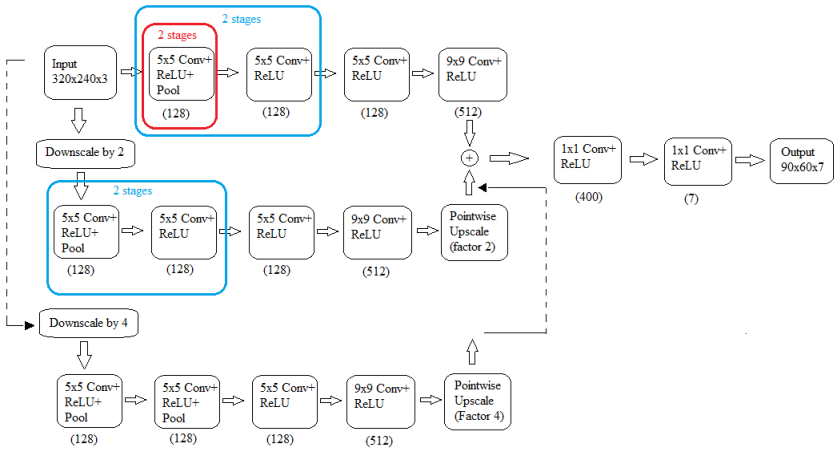
Figure 4. Architecture of Network 2



3.3 Network 3

We found that Network 2 performs better than Network 1. But, both these networks underperform the baseline network. Based on these observations, the next natural thing to do is to make the baseline model deeper. In our experiments, when we made both the feature extraction layers and the fully connected layers (which are modelled as $1X1$ convolutions) deeper, we got a marginal dip in the performance. One possible reason to explain this observation is that, due to an increase in the number of parameters, the network may be overfitting. So, a careful increase in the depth has to be made. In the Network 3, we perform a depth increase only on the feature extraction layers. Moreover, this depth increase is similar to that in Network 2, where high resolution banks get more increase in the depth as compared to lower resolution banks. The architecture of Network 2 is shown in Figure 5.

Figure 5. Architecture of Network 3



4 Results and Discussion

We trained our models using the FLIC dataset, which is a labelled dataset of human poses, with the positions of the body joints annotated. Training is performed using Mean Square Error loss function, with target outputs as n heat-maps, where each heat-map is a Gaussian centred around the ground truth joint location. We used batch Gradient Descent with batch size 8 for training the networks.

For evaluation of the test-set performance we use the measure suggested by Sapp et. al. [4]. For a given normalized pixel radius (normalized by the torso height of each sample) we count the number of images in the test-set for which the distance of the predicted joint location to the ground-truth location falls within the given radius.

The performance of the baseline approach having three resolution banks is shown in Figure 6. We observe that having a spatial model on top of the part detector improves the performance. This improvement is marginal as the part detector by itself gives very good

performance. However, in our experiments, we observed that spatial model gives better improvements when used on top of 1 resolution bank, which by itself performs bad.

Figure 6. Baseline performance (Left Wrist)

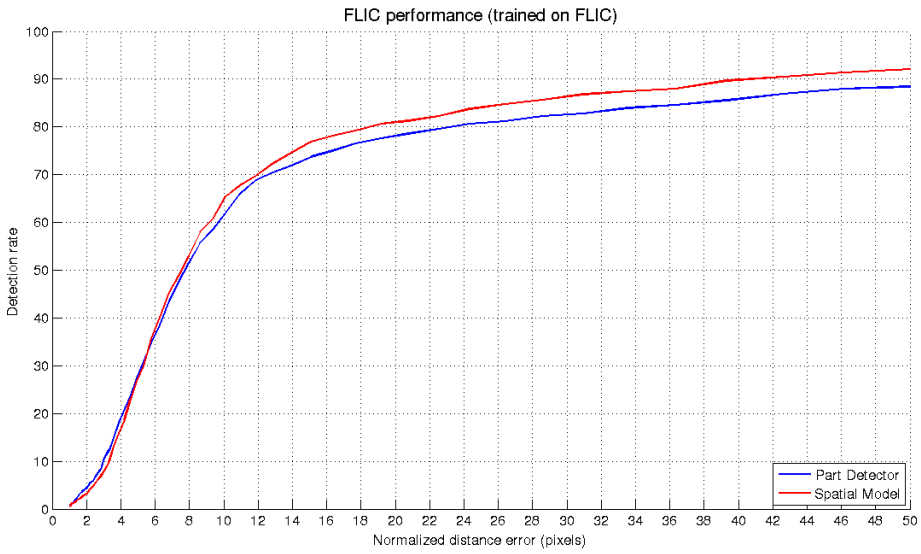


Figure 7 shows the performance of the part detector of the baseline model with varying number of resolution banks. We observe that more the number of resolution banks, better is the performance. This is because with multiple resolution banks, we capture finer details along with the contextual information, and by suitable combining both, we need good performance boost. Since 3 resolution bank network performs the best, we used three banks for the rest of the experiments.

Figure 8 is a comparison of the performance of the proposed Networks with the baseline approach. We observe that Network 1 and Network 2 significantly underperforms the baseline Network. However, Network 2 performs better than Network 1. This is an indication of bias (ie) less number of parameters in Network 1 than what is needed. Network 3, on the other hand outperforms the baseline approach. This validates our hypothesis that increasing the depth of the network according to the resolution of the bank would improve the performance.

Figure 9 shows the predictions obtained on some of the test cases. The heat-map predicted by the network represents the probability that a part is found in a given location. We take the location of the maximum value in this heat-map as the predicted joint location. The predictions are colour coded as follows: red represents shoulders, green represents elbow, blue represents wrist and yellow represents face. Figures 9(a) and 9(d) are some examples of good predictions of the part detector. The test cases where part detector fails are shown in Fig 9(b) and 9(c). In 9(b), we observe that both the left and the right wrists are predicted in the same location, whereas in 9(c), the left wrist is wrongly predicted. Figures 9(e) and 9(f) show the improvements brought by the spatial model when these wrong predictions are fed to it. Both the wrist locations are predicted properly after getting passed through the spatial model.

Figure 7. Performance of part detector (on left wrist) with number of resolution banks

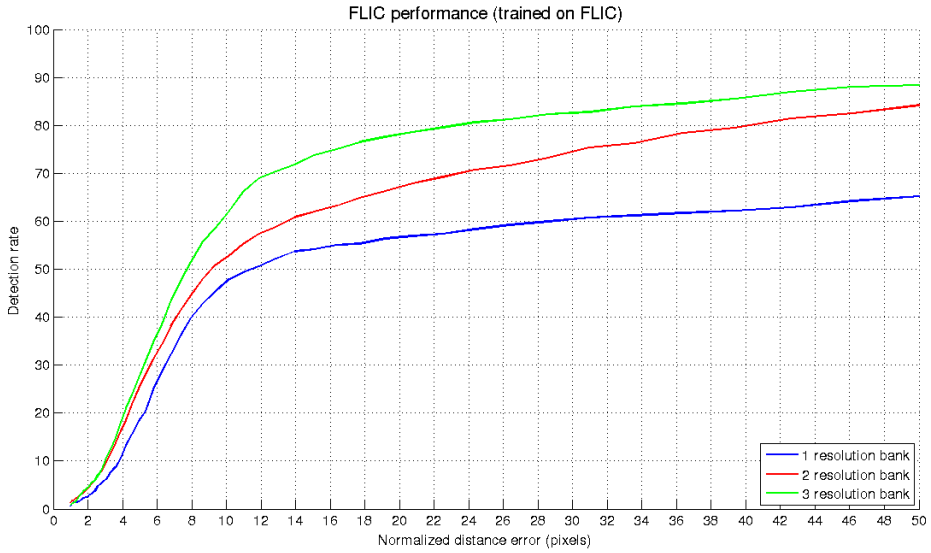


Figure 8. Comparison of proposed approaches (on left wrist) with the baseline model

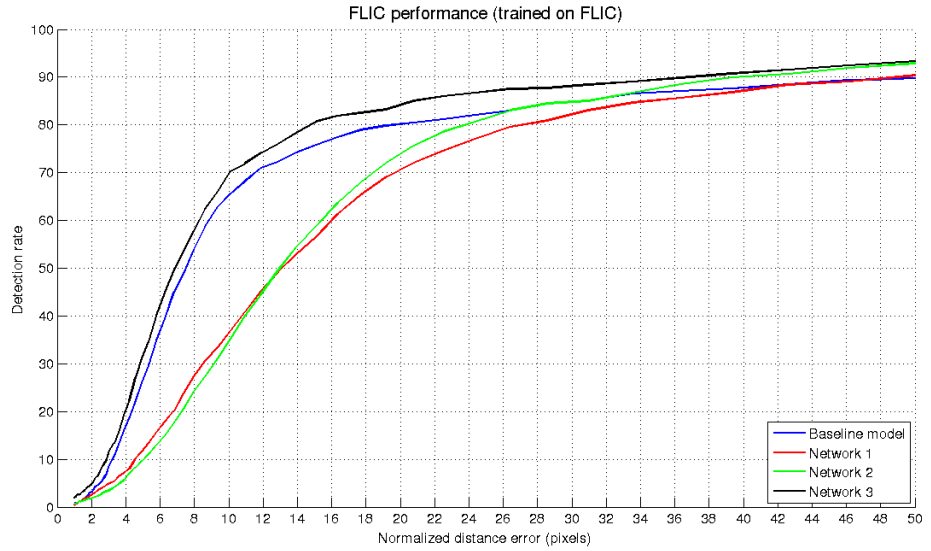


Figure 9. Some Predicted heat-maps

(a) Part Detector Output

(b) Part Detector Output

(c) Part Detector Output

(d) Part Detector Output

(e) Spatial Model Output

(f) Spatial Model Output

5 Conclusion

In this project, we implemented the Pose Estimation problem in Deep Learning framework and showed that this performs very well. We experiment with different network architectures and finally came up with an architecture (Network 3) that achieves marginal improvements compared to the baseline approach.

References

[1] Jonathan Tompson, Arjun Jain, Yann LeCun, and Christoph Bregler. *Joint Training of a Convolutional Network and a Graphical Model for Human Pose Estimation.* In NIPS 2014.

[2] K. Simonyan and A. Zisserman. *Very deep convolutional networks for large-scale image recognition.* In ICLR 2015.

[3] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovic. *Going deeper with convolutions.* In CVPR 2015.

[4] B. Sapp and B. Taskar. *Modec: Multimodal decomposable models for human pose estimation.* In CVPR 2013.