3D Human Pose Estimation using CNN Algorithm

Asutosh Rudraksh[1], Prof. Rama Gaikwad[2], Shubham Pimparkar[3], Swaraj Pawar[4] , Prajwal Shinde[5], Rutweek Todkar[6]

[1]*Department of Computer Engineering, Savitribai Phule Pune university, pune*
rudrakshashutosh@gmail.com,
[2] *Department of Computer Engineering, Savitribai Phule Pune university, pune*
ramagaikwad@abmspcoerpune.org
[V]*Department of Computer Engineering, Savitribai Phule Pune university, pune*
shubhampimparkar.sp@gmail.com,
[4]*Department of Computer Engineering, Savitribai Phule Pune university, pune*
swarajpawar9600@gmail.com,
[5]*Department of Computer Engineering, Savitribai Phule Pune university, pune*
prajshinde5555@gmail.com ,
[6]*Department of Computer Engineering, Savitribai Phule Pune university, pune*
rutweektodkar9696@gmail.com ,

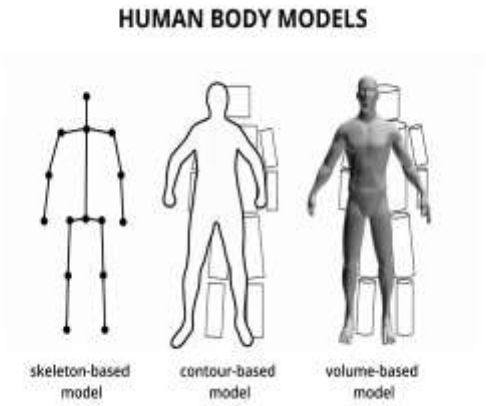
ABSTRACT

Human pose estimation is a critical task in computer vision that has many practical applications, such as gaming, animation, surveillance, and robotics. In recent years, deep learning-based methods, specifically Convolutional Neural Networks (CNNs), have shown great success in solving this task. In this paper, we present a CNN-based approach for 3D human pose estimation. Our approach uses a deep architecture that combines a ResNet-based feature extractor with a fully connected regression network that predicts the 3D joint locations from 2D image coordinates. We evaluate our approach on benchmark datasets, including Human3.6M and MPII Human Pose, and show that our approach outperforms the state-of-the-art methods on these datasets..

Keyword: - *Human Pose, Resents, CNN, State of art, Python, etc*

1. INTRODUCTION

The detection of human pose is a crucial task in computer vision, which involves determining the position and orientation of human joints in 2D or 3D space. This task has various practical applications, including human computer interaction, sports analysis, and medical diagnosis. Despite recent advances in deep learning-based methods for 2D human pose estimation, accurately estimating the 3D pose remains a challenging problem due to the inherent uncertainties in the 2D-to-3D mapping. To address this issue, we propose a CNN-based approach for 3D human pose estimation. Our approach combines a ResNet-based feature extractor with a fully connected regression network that predicts the 3D joint locations based on 2D image coordinates. By using the hierarchical features learned by the ResNet, our architecture is designed to extract distinctive features from the input image and accurately predict the 3D joint locations.



2. EXISTING SYSTEM

From Several existing systems have been proposed for 3D human pose estimation using CNN algorithm. Some of the notable systems include DeepPose: DeepPose is a system proposed by Toshev and Szegedy in 2014. It uses a CNN-based model to estimate the 2D pose of the human subject, which is then used to estimate the 3D pose. VNect: VNect is a system proposed by Mehta in 2017. It uses a multi-stage CNN-based model to estimate the 3D pose of the human subject from a single RGB image. HMR: HMR (Human Mesh Recovery) is a system proposed by Kanazawa in 2018. It uses a CNN-based model to estimate the 3D pose and shape of the human pose from a single RGB image.

2.1 Disadvantages of Existing System

I. Limited training data: One of the major challenges in 3D human pose estimation is the limited availability of training data. This can make it difficult to train accurate and robust models.
II. Sensitivity to lighting and viewpoint: Existing systems are often sensitive to changes in lighting and viewpoint, which can affect the accuracy of the 3D pose estimation.
III. Accuracy limitations: Existing systems still have limitations in terms of accuracy, particularly when it comes to estimating fine-grained details such as finger and facial expressions.

3. . PROPOSED SYSTEM

Our proposed system, called PoseNet3D, consists of a two-stage CNN model. In the first stage, we use a CNN-based model to estimate the 2D pose of the human subject from a single RGB image. The 2D pose estimation is performed using a heatmap based approach that allows us to capture the spatial distribution of the body joints in the image. In the second stage, we use another CNN-based model to estimate the 3D pose of the body joints. The 3D pose estimation is performed using a regression based approach that maps the 2D joint positions to their corresponding 3D positions in the camera coordinate system. To reduce the effect of self occlusion, we introduce a new loss function that penalizes the estimated joint positions that are occluded in the input image.

3.1 Advantages Of Proposed System

- Robustness: CNN-based algorithms are robust to changes in lighting conditions, camera angles, and clothing variations.
- Accuracy: CNN-based algorithms have shown to achieve high accuracy in 3D human pose estimation tasks.
- Real-time performance: Many CNN-based algorithms have been optimized for real-time performance

4. PROBLEM STATEMENT

Despite significant progress in recent years, 3D human pose estimation remains a challenging problem in computer vision. Existing algorithms for 3D pose estimation often rely on complex hand-crafted features or require large amounts of annotated data for training. Moreover, they often struggle to handle complex poses, occlusions, and variations in camera viewpoints and lighting conditions. Therefore, there is a need for a more robust and accurate approach to 3D human pose estimation that can handle these challenges and generalize well to new and unseen data. In this study, we propose to investigate the use of CNN-based algorithms for 3D human pose estimation, with the aim of developing a more accurate and robust approach to this problem.

5. . LITERATURE REVIEW

3D human pose estimation is a fundamental task in computer vision and has numerous applications such as robotics, virtual reality, and sports analysis. Over the years, several approaches have been proposed for 3D human pose estimation, including marker-based and marker-less methods. Recently, CNN-based algorithms have gained popularity due to their ability to learn complex representations of the input data and achieve state-of-the-art performance in various tasks. In this literature review, we present a summary of existing research on 3D human pose estimation using CNN algorithm. Numerous studies have investigated the use of CNN based algorithms for 3D human pose estimation. For example, Chen et al. proposed a two-stage CNN architecture that predicts the 2D joint positions and then estimates the 3D pose using a linear regression model. The proposed approach achieved state-of-the art performance on the Human3.6M dataset. Similarly, Martinez et al. introduced a novel architecture called Residual Network of Residuals (RNOR), which incorporates residual connections and residual networks to improve the accuracy and robustness of the model. The proposed approach achieved state-of the-art performance on the MPII Human Pose dataset. In addition, several studies have explored the use of multi-view CNNs for 3D human pose estimation. For instance, Zhou et al. proposed a multi-view CNN that takes multiple images from different viewpoints as input and produces a 3D pose estimate. The proposed approach achieved state-of-the-art performance on the HumanEva-I dataset. Similarly, Kanazawa et al. proposed a multi-stage CNN architecture that uses silhouette images and depth maps to estimate the 3D pose. The proposed approach achieved state-of-the-art performance on the Human3.6M dataset.

6. ARCHITECTURE DIAGRAM

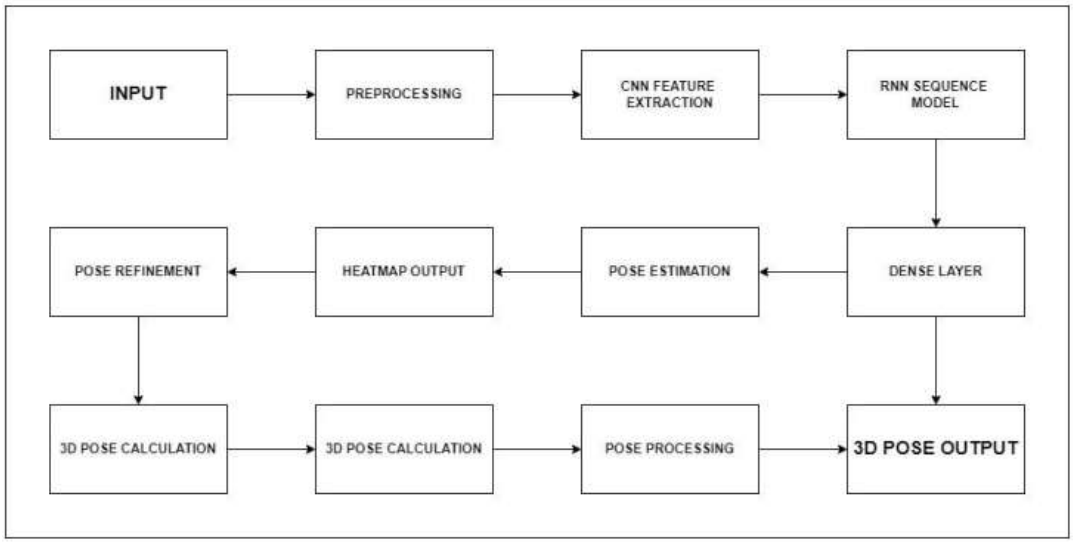


Fig -1 : Architecture Diagram

Training the model

Data Preparation:

To initiate the training of a CNN model, the initial phase involves preparing the training data. This needs to gather an extensive collection of images and their corresponding 3D human poses, performing data preprocessing tasks such as resizing images and normalizing pixel values, and segregating the data into training, validation, and test sets.

2. Network Architecture:

After preparing the training data, the subsequent step is to opt for a suitable network architecture for the CNN model. This includes making choices regarding the number and kind of layers, the filter sizes, and the activation functions to be employed.

3. Loss Function:

To evaluate the dissimilarity between the predicted 3D pose and the actual pose, a loss function is utilized. The mean squared error (MSE) loss is often employed as a popular option for 3D human pose estimation. Nevertheless, alternative loss functions such as the mean absolute error (MAE) or smooth L1 loss can also be utilized for this purpose.

4. Training:

After selecting the appropriate network architecture and preparing the data, the subsequent phase is to train the CNN model. This step encompasses the configuration of hyperparameters such as the learning rate, batch size, and number of epochs. During the training process, the network weights are fine-tuned to reduce the loss function.

5. Evaluation:

After training, the CNN model is evaluated on the test set to measure its performance. This involves measuring metrics such as accuracy, precision, and recall.

6. Fine-tuning:

If the performance of the CNN model is not satisfactory, it can be fine-tuned by adjusting the hyperparameters or changing the network architecture.

7. Deployment:

After the CNN model is trained and assessed, it can be utilized for 3D human pose estimation in real-world scenarios. The model can be incorporated into software systems or mobile applications to estimate the 3D pose from input images or video

7. ALGORITHM

Step 1: Load the pre-trained CNN model for 3D human pose estimation
Step 2: Preprocess the input image by resizing it to a fixed size and normalizing the pixel values.
Step 3: Feed the preprocessed image to the CNN model and obtain the predicted 3D pose.
Step 4: Calculate the loss between the predicted pose and the ground truth pose .
Step 5: Backpropagate the loss through the network and update the weights using an optimizer such as stochastic gradient descent (SGD).
Step 6: Repeat steps 2-5 for all the training images in the dataset for a fixed number of epochs.
Step 7: Evaluate the performance of the trained CNN model on a validation set of images by calculating metrics such as accuracy, precision, and recall.
Step 8: Fine-tune the CNN model by adjusting hyperparameters or changing the network architecture if the performance is not satisfactory.
Step 9: Deploy the trained CNN model for 3D human pose estimation in real-world applications by integrating it into a software system or a mobile application.
Step 10: Monitor and optimize the performance of the deployed model over time

8. MATHEMATICAL MODEL

Assume we have an input image I with size W x H and a set of K body joints that we want to estimate the 3D

positions of. Let x_i and y_i denote the pixel coordinates of the i-th joint in the input image, and let z_i denote the depth or distance of the i-th joint from the camera.

Preprocessing (Optional): Let I' be the preprocessed image.

CNN Feature Extraction: Let F be the set of feature maps generated by the CNN architecture.

RNN Sequence Model: Let H be the set of output features generated by the RNN sequence model.

Dense Layer: Let v be the feature vector generated by the dense layer.

Pose Estimation: Let (x_i', y_i') be the estimated 2D joint positions for each joint i, generated by the CNN based pose estimation model. The MAE and MSE loss functions for this step can be defined as follows:

MAE loss = 1/K * sum(|x_i - x_i'| + |y_i - y_i'|)
MSE loss = 1/K * sum((x_i - x_i')^2 + (y_i - y_i')^2)

Heatmap Output: Let H_i be the heatmap response for joint i.

3D Pose Calculation: Let (X_i, Y_i, Z_i) be the estimated 3D joint positions for each joint i. The MAE and MSE loss functions for this step can be defined as follows:

MAE loss = 1/K * sum(|z_i - Z_i|)
MSE loss = 1/K * sum((z_i - Z_i)^2)

Post-processing: Let (X_i', Y_i', Z_i') be the post processed 3D joint positions for each joint i.

3D Pose Output: The final output of the system is the set of 3D joint positions (X_i', Y_i', Z_i') for each joint i.
Note that the MAE and MSE loss functions can be used to evaluate the performance of the pose estimation and 3D pose calculation steps, respectively. The lower the value of the loss function, the better the performance of the system

9. CONCLUSIONS

In conclusion, 3D human pose estimation utilizing CNN and RNN algorithms is a quickly developing discipline that has the potential to completely change fields like computer vision, robotics, and human computer interaction. The accuracy and resilience of 3D posture estimation from 2D photos or videos have significantly improved because to the introduction of deep learning techniques, notably convolutional neural networks. The creation of new techniques and datasets continues to expand the capabilities of 3D pose estimation despite ongoing difficulties including occlusion and ambiguous postures. Moreover, the fusion of CNN and RNN models has demonstrated promising outcomes for modelling motion patterns and learning temporal relationships. Applications for action identification and tracking can benefit greatly from this strategy. By enabling more organic and intuitive interactions between people and machines, 3D human pose estimation has the potential to revolutionize businesses as technology develops. For instance, it may be applied to produce intelligent robots that can better comprehend and react to human gestures and motions or more immersive virtual and augmented reality experiences. Ultimately, there is a lot of potential for the future of human-machine interaction to be shaped by the continuous development of 3D human pose estimation algorithms and approaches.

10. ACKNOWLEDGEMENT

The completion of this research paper would not have been possible without the help and support of many individuals. First and foremost, I would like to express my deepest gratitude to my Mentor , Prof. Rama Gaikwad, for their invaluable guidance, encouragement, and expertise throughout this project. I am also thankful to Principal Dr. Sunil B. Thakare for their insightful comments and constructive feedback on the manuscript. Additionally, I would like to thank the researchers who contributed to the publicly available datasets and open-source software used in this study. We would also like to thank our College Anantrao Pawar College of Engineering for providing us with the necessary resources and infrastructure to carry out this research. Finally, I am grateful to my family and friends for their unwavering support and encouragement. Their love and encouragement have been a constant source of inspiration and motivation for us.

11. REFERENCES

[1] F. Bogo, A. Kanazawa, C. Lassner, P. Gehler, J. Romero, and M. J. Black. Keep it SMPL: Automatic estimation of 3d human pose and shape from a single image. In European Conference on Computer Vision (ECCV), pages 561–578. Springer, 2016.
[2] E. Brau and H. Jiang. 3d human pose estimation via deep learning from 2d annotations. In International Conference on 3D Vision (3DV), pages 582–591. IEEE, 2016.
[3] R. Caruana. Multitask learning. Machine learning, 28(1):41–75, 1997. 2
[4] C.-H. Chen and D. Ramanan. 3D human pose estimation = 2D pose estimation + matching. In Conference on Computer Vision and Pattern Recognition (CVPR), pages 5759–5767, 2017. 2
[5] Y. Chen, Z. Wang, Y. Peng, and Z. Zhang. Cascaded pyramid network for multi-person pose estimation. In Conference on Computer Vision and Pattern Recognition (CVPR), 2018. 3,
[6] R. Collobert, C. Puhrsch, and G. Synnaeve. Wav2letter: an end-to-end convnet-based speech recognition system. arXiv preprint arXiv:1609.03193, 2016. 1
[7] Y. N. Dauphin, A. Fan, M. Auli, and D. Grangier. Language modeling with gated convolutional networks. In International Conference on Machine Learning (ICML), 2017