CS230

Predicting the Gait Deviation Index of Children with Cerebral Palsy

Vincent J. Salpietro, Jon P. Stingel, and Cara G. Welker
CS 230 Final Report
Stanford University
vsal@stanford.edu, stingjp@stanford.edu, cgwelker@stanford.edu

Abstract

Gait deviation index (GDI) is a comprehensive clinical measure of walking used to help inform treatment decisions for children with cerebral palsy (CP). Currently, this score is calculated with motion capture, which is expensive and requires specialized spaces and trained personnel. In this paper, we aimed to use deep learning techniques to predict GDI with 2D video data instead. Our dataset consisted of approximately 500 videos of CP patients walking at Gillette Children's Hospital paired with GDI scores computed from motion capture collected in the same visit. These were run through the DensePose network, and then trained an additional network that combined spatial and temporal features to classify the patient's motion into a specific GDI score range. In validation, the correlation between predicted and true GDI scores was 0.568. This is lower than the current state-of-the-art correlation of 0.74, but proves to be promising. The state-of-the-art network utilized significantly more data, and so the model generated in this study provides a solid foundation to expand with more data, and further refinement of hyperparameters and architecture.

1 Introduction

About 3 out of every 1000 children in the United States are born with Cerebral Palsy (CP) [Arneson et al, 2009]. Cerebral Palsy results from damage to the developing brain and can affect the way people move and control their muscles. Although proper surgery can significantly improve patients' ability to walk, many children undergo surgery with no improvement because there is no standardized method for predicting which surgeries will be helpful or even quantifying if their gait is improving or deteriorating. Recently, an overall quantitative assessment of gait pathology called the gait deviation index (GDI) has been proposed to mitigate some of these issues [Schwartz et al, 2008]. This index measures how far away certain kinematic parameters are from normal throughout a gait cycle, and requires whole-body motion capture. This is expensive (on the order of tens to hundreds of thousands of dollars) and requires specialized training - usually someone with knowledge of physiology and anatomy to correctly utilize motion capture methods and an engineer to analyze the results. If this could be estimated using raw video data, significant amounts of time and money would be saved, and patients would be able to better monitor their progress simply by taking videos from home. Through this project, the focus was to predict GDI using videos of patients walking with CP that could be taken with any camera.

CS230: Deep Learning, Winter 2018, Stanford University, CA. (LateX template borrowed from NIPS 2017.)

2 Related work

2.1 Image Classification

Image classification has been proven to be a particular task where deep learning can excel. Network architectures such as AlexNet [Krizhevsky et al, 2008], VGG16 [Simonyan et al, 2015], and ResNets [He et al, 2015] have proven that they are capable of properly classifying details about an image. Many of these architectures are able to classify millions of different images of various objects correctly, as they are tested using a database called ImageNet that contains over 14 million labeled images. The resulting networks have been used for many tasks from facial recognition to autonomous driving, and can be expanded and used for other tasks involving images in the medical and health related fields.

2.2 Classification in Videos

Although the deep learning community has had significant success with image classification, action recognition in videos has had much slower progress [Ghosh, 2018]. Large computational cost, the need to capture context over periods of time, and no defined 'best' architecture are several reasons for the slow development. Existing video classification architectures differ in how they combine spatial and temporal components, some architectures choosing to implement separate spatial and temporal streams (two stream networks) [Simonyan et al, 2014], while others choosing to input consecutive frames that get fused at different layers of the network (single stream networks) [Ramasinghe et al, 2015]. One well performing implementation of video activity recognition uses a long-term recurrent convolutional network, which first converts sequential images to image embeddings using a convolutional neural network (CNN), and then feeds these into a long short-term memory network (LSTM) to output a single prediction [Donahue et al, 2015].

2.3 Gait Prediction

Recently, deep learning techniques have resulted in more accurate video-based motion tracking through various software packages, such as estimation of up to 135 key points on a human skeletal frame in OpenPose [Cao et al, 2017] or even estimation of the entire 3D surface of a human body in DensePose [Guler et al, 2018]. Previous work has used deep learning to correlate the outputs of OpenPose with GDI, using 3,000 videos of children with CP walking from Gillette Children's Hospital, both with and without motion capture [Kidzinski et al]. These results from the skeletal estimations of OpenPose were promising (correlation = 0.74 with motion capture calculations of GDI), but could potentially be improved using the 3D surface estimation of DensePose. This study focuses on utilizing a subset of the same dataset to correlate DensePose outputs of children walking with cerebral palsy with their GDI, in the hopes of improving predictions.

3 Dataset

The data used for this study comes from Gillette Children's Hospital and consists of approximately 500 videos of children with cerebral palsy walking, each video corresponding to one gait lab visit by one patient. In addition, both the frontal and sagittal views of the patient throughout the task are represented in the same video file. During each gait lab visit, the walking videos are paired with motion capture to quantify the gait characteristics of the child. The results from motion capture can be used to calculate a GDI score for the left and right leg of each child, which is paired with the video of the patient visit for that day. This provides each of the videos in which motion capture data is available with a "ground truth" value of GDI, as motion capture is the state-of-the-art method of computing GDI. For privacy purposes, direct access to the raw patient videos was not granted for this study. However, access to a subset of the database (approximately 500 videos) after being passed through the DensePose network was obtained. This network (DensePose) takes in 2-dimensional images and outputs a surface mesh of the subject with depth encoded, as shown in Figure 1. The DensePose output image incorporates depth and an estimation of the patient's body through a specific color scheme. When an image or video is run through this network, it outputs two images - one encoding body segmentation and curvature of the surface (480 x 640 x 3), and one encoding key points that outline the human body (480 x 640 x 1). For this study, the former was used as it captured more information about the patient's pose and shape.

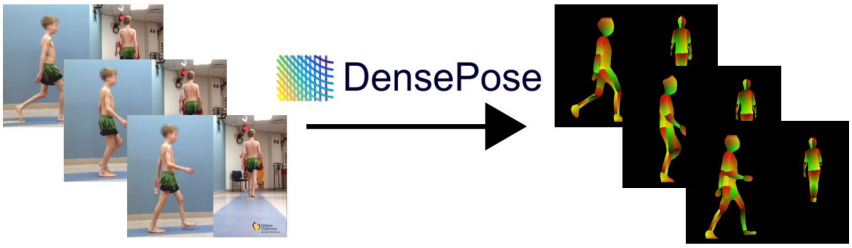


Figure 1: Video frames were passed through the DensePose network in order to encode the body mesh into an output image.

3.1 Data Management

Gathering and organization of the custom dataset required a significant amount of time and effort. The DensePose outputs contained two images for each video frame as described above; however, only the image containing body segment and curvature encodings was used in this study. Therefore, the database had to be filtered such that only the desired type of images were paired together in the correct sequence to form each video. When we received the data, all frames from the same video were grouped into a common directory labeled with the video ID number. The ground truth GDI labels for each video from the motion capture data were stored in a large .csv file provided for the entire database of videos (including those the study did not have access to). Video ID numbers were then used to link video file locations with the corresponding GDI values (a focus only on right leg GDI for the purposes of this study to prove feasibility). By referencing only the video ID numbers that existed in the file directory, all videos that the team's data base did not contain were filtered out, leaving only videos that had a calculated GDI label.

The team originally expected to obtain approximately 3,000 labeled videos, the whole database run through DensePose. Due to technical issues with DensePose, the final dataset reported on in this study is only a subset of about 500 videos with labels. This was split in an 80:10:10 manner to create a training, testing, and validation sets. In order to augment the small amount of data, each video was divided into smaller video segments of 124 frames (full videos average over 500 frames each). Segments were spaced throughout the length of the full video, with starting indices spaced 31 frames apart. Each smaller video segment was paired with the label corresponding to the original full video, giving us a total of about 6500 124-frame videos. All of the augmented data was isolated in its respective split to prevent biasing the model (i.e. all segments from the same full video were in the same set).

4 Methods

In order to handle the temporal nature of the input gait videos, a mixed convolutional (CNN) and sequential (LSTM) model was constructed from a similar architecture used for activity recognition [Donahue et al, 2015]. The final architecture design can be found in Figure 2.

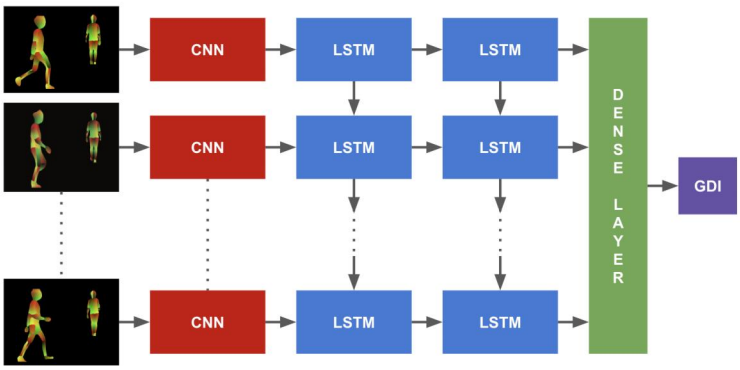


Figure 2: Architecture consisting of a CNN stacked with an LSTM sequential network.

4.1 CNN

Convolutional Neural Networks (CNN) are often used in image-related learning tasks as they can effectively pull out features from visual data. This implementation utilizes a popular architecture called Alexnet, which uses 8 main layers [Krizhevsky et al, 2012]. Alexnet pre-trained weights from the Imagenet dataset were used, and images were resized from 480x640x3 to 227x227x3 in order to fit into the model architecture, and decrease computational time. Since this network was not used directly for predicting classes, the last layer of the network was dropped. Processing all of the images through the first 7 layers of the network then produces a 4096 length feature vector for each video frame. In addition to AlexNet, a custom CNN model that consisting of 6 layers instead of 8 was constructed and tested. All layers of this network were trained based on the frame's corresponding video GDI, its input was reduced to a size of 64x64x3 in order to decrease computational cost.

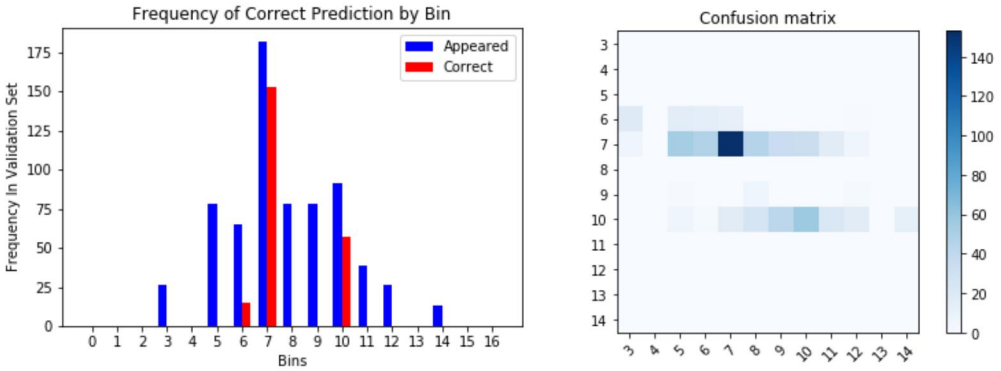
4.2 LSTM

LSTM is sequential network architecture used for recurrent neural networks in order to provide context to data over time. In this model, each LSTM unit takes in a single image frame as an input and uses contextual information provided by the surrounding units in order to make a prediction. This is fitting for gait video data, as GDI is calculated based on deviations from normal gait kinematics at that time in the gait cycle. In order to make good predictions on a small dataset, the model was set up to bin GDI scores such that the model had to classify the video as one of the binned GDI score ranges. The categorical cross-entropy loss function was used to help drive weights in mini-batch gradient descent, as seen in Equation 1. This loss function rewards the model for predicting the correct bin, and penalizes it for predicting any other bin. Bins were set to span 5 GDI points, a clinically significant difference in score [Schwartz et al, 2008]. Because the dataset included GDI scores from 35 to 120, 17 bins were created.

$$\mathscr{L}(\hat{y}, y) = \sum_{i}^{N} [y_i log \hat{y}_i + (1 - y_i) log(1 - \hat{y}_i)] \tag{1}$$

5 Results and Discussion

Accuracy and correlation from our predicted GDIs in the validation set to the GDI labels from motion capture were calculated. Correlation was used to compare performance to the only other known network that has attempted this problem, which used correlation as their metric. This other approach had a large dataset and used a regression approach with a correlation metric [Kidzinski et al]. The majority of time spent initially on this project was searching for the best architecture for the problem. As a result, hyperparameters such as batch size and learning rate were not varied. Throughout all architectures, our learning rate was 0.001, our mini-batch size was 512 in order to maximally speed up computation without running out of memory. Xavier initialization was used for model initialization.



(a) Frequency of target appearance vs. correct predictions of LSTM model on validation set

(b) Confusion matrix for our LSTM model on the validation set

Figure 3: Our LSTM model was able to predict GDI scores in bins 6, 7, and 10

Table 1: Summary statistics comparing different models run on the DensePose dataset and showing improvement from the baseline CNN to the addition of LSTM model.

Model	Val Acc	Correlation
CNN Baseline	19.8%	0.331
Pretrained Alexnet	25.8%	
Finetuned Alexnet	28.5%	0.202
Alexnet + LSTM	**33.3%**	**0.568**

All models were trained on 10 epochs. Hyperparameters such as number of layers to retrain in the AlexNet were varied. The team examined the performance of AlexNet on predicting GDI (without LSTM) on individual frames of videos while retraining just the last layer (8) and the last two layers (7,8). The team ultimately used the pre-trained weights due to minimal performance increases. The minimal increase reaffirmed that the use of some sequential model would be needed in order to maximize performance.

Our model with the LSTM performed better than the three purely CNN architectures explored (See Table 1). This result was expected due to the temporal aspect of GDI. The comparison between the AlexNet and CNN Baseline architectures is unclear because AlexNet had higher accuracy, but CNN baseline had a better correlation. The features that each of the CNNs is learning to detect could be very different, as the baseline was trained directly on this dataset, and the AlexNet was trained on millions of regular images.

When looking at the predictions for the AlexNet and LSTM paired model, it appeared that the network learned to predict several select bins (See Figure 3), and group all GDIs in the surrounding area around these bins. This grouping is likely a result of the small training set (500 independent videos and labels). Although we augmented our training set by splitting these videos into 13 smaller clips each, this doesn't help to change the distribution of the GDI labels in the data set. The model tended to group predictions into bins 6,7 and 10, which are some of the most frequent scores that occur in the dataset. This could potentially be mitigated by normalizing the frequency with which each score occurs, or by weighting the importance of the classes to drive learning of more features for the other classes.

6 Conclusion and Future Work

The problem of predicting GDI from 2D videos is a difficult one, and there are many different architectures and parameters that can be investigated to solve it. Compared to previous work feeding in x,y data points from OpenPose to predict the same metric, our input of DensePose images is more computationally expensive, but has the potential to improve predictions or decrease the size of training set needed. After some initial exploration of different network architectures and hyperparameters, we were not able to achieve as high of a correlation between the predicted and true GDI as the previous OpenPose model (0.57 versus 0.72). However, having used a significantly smaller dataset (500 versus 3000 labeled videos) we believe that this approach still could be improved. A combined CNN and LSTM performed much better than a CNN alone, and sets a solid model to refine parameters.

Several strategies will be examined moving forward in this work. An end-to-end model may significantly increase performance. Currently, the CNN portion of the model was trained on a one-to-one image to classification relationship, whereas the objective of our study was develop a relationship of a sequence of images to a single classification. Therefore, the features that the CNN model is selecting and passing on in the image embeddings may not be helpful for the LSTM. Retraining the CNN directly attached to the LSTM layers will allow us to train both parts of the model under the desired objective, such that features selected in the image convolution will be useful for the LSTM layers later. In addition, training a custom CNN that takes in our images without downsizing could produce better results, as resizing of the images may influence the encodings that are output from DensePose. Finally, the team is working to gain access to the full dataset which will grant more flexibility. With a larger dataset, strategies such as regression could be more feasible.

7 Contributions

All team members worked collaboratively throughout the project. Jon was primarily responsible for processing and formatting the data, as well as being instrumental in debugging. Cara was responsible for setting up data logs as well as the point person for both the paper and poster. Vincent was responsible for the model architecture search and implementation, and finding that one statistic about CP.

References

[1] Arneson et al (2009) Prevalence of Cerebral Palsy: Autism and Developmental Disabilities Monitoring Network. *Disability and Health Journal* **2**(1):45-48.

[2] Cao et al (2017) Realtime multi-person 2D pose estimation using part affinity fields. *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*.

[3] Donahue et al (2015) Long-term recurrent convolutional networks for visual recognition and description. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* **39**(4):677-691.

[4] Ghosh, R. (2018) Deep Learning for Videos: A 2018 Guide to Action Recognition. *Qure.ai Blog*, http://blog.qure.ai/notes/deep-learning-for-videos-action-recognition-review

[5] Guler et al (2018) DensePose: Dense Human Pose Estimation In The Wilds. *arXiv*.

[6] He et al (2015) Deep Residual Learning for Image Recognition. *CoRR* **1512.03385**

[7] huanzhang12 Tensorflow Alexnet Model https://github.com/huanzhang12/tensorflow-alexnet-model

[8] Kidzinski et al Automatic diagnostics of gait pathologies using a mobile phone. *unpublished work*

[9] Krizhevsky et al (2012) ImageNet Classification with Deep Convolutional Neural Networks. *NIPS 2012*.

[10] Ramasinghe et al (2015) Action recognition by single stream convolutional neural networks: An approach using combined motion and static information. *2015 3rd IAPR Asian Conference on Pattern Recognition (ACPR)*101-106

[11] Schwartz et al (2008) The gait deviation index: A new comprehensive index of gait pathology. *Gait and Posture* **28**(3):351-357.

[12] Simonyan et al (2014) Very Deep Convolutional Networks for Large-Scale Image Recognition. *CoRR* **1409.1556**

[13] Simonyan et al (2014) Two-Stream Convolutional Networks for Action Recognition in Video. *CoRR* **1406.2199**

Github Repository

https://github.com/cgwelker/clinical-parameters-gait