Computer Vision for Gait Assessment in Cerebral Palsy: Metric Learning and Confidence Estimation

Peijun Zhao, Moises Alencastre-Miranda, Zhan Shen, Ciaran O'Neill, David Whiteman, Javier Gervas-Arruga, and Hermano Igo Krebs, *Fellow, IEEE*

***Abstract*— Assessing the motor impairments of individuals with neurological disorders holds significant importance in clinical practice. Currently, these clinical assessments are time-intensive and depend on qualitative scales administered by trained healthcare professionals at the clinic. These evaluations provide only coarse snapshots of a person's abilities, failing to track quantitatively the detail and minutiae of recovery over time. To overcome these limitations, we introduce a novel machine learning approach that can be administered anywhere including home. It leverages a spatial-temporal graph convolutional network (STGCN) to extract motion characteristics from pose data obtained from monocular video captured by portable devices like smartphones and tablets. We propose an end-to-end model, achieving an accuracy rate of approximately 76.6% in assessing children with Cerebral Palsy (CP) using the Gross Motor Function Classification System (GMFCS). This represents a 5% improvement in accuracy compared to the current state-of-the-art techniques and demonstrates strong agreement with professional assessments, as indicated by the weighted Cohen's Kappa ($\kappa_{lw} = 0.733$). In addition, we introduce the use of metric learning through triplet loss and self-supervised training to better handle situations with a limited number of training samples and enable confidence estimation. Setting a confidence threshold at 0.95, we attain an impressive estimation accuracy of 88%. Notably, our method can be efficiently implemented on a wide range of mobile devices, providing real-time or near real-time results.**

Manuscript received 1 November 2023; revised 18 March 2024 and 5 May 2024; accepted 28 May 2024. Date of publication 18 June 2024; date of current version 1 July 2024. This work was supported by the Takeda Development Center Americas, Inc. (successor in interest to Millennium Pharmaceuticals, Inc.) MIT under Grant #6947514. An earlier version of this paper was presented in part at the IEEE-EMBS International Conference on Body Sensor Networks: Sensor and Systems for Digital Health (IEEE BSN 2023) [DOI: 10.1109/BSN58485.2023.10331472]. *(Corresponding author: Peijun Zhao.)*

Peijun Zhao was with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA. He is now with the Global Technology Applied Research, JPMorganChase, New York, NY 10017 USA (e-mail: peijun.zhao@jpmchase.com).

Moises Alencastre-Miranda, Ciaran O'Neill, and Hermano Igo Krebs are with the Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: moisesam@mit.edu; ciarnoneill@mit.edu; hikrebs@mit.edu).

Zhan Shen was with the Robotics Institute, University of Michigan, Ann Arbor, MI 48109 USA. He is now with the Sofia University, Palo Alto, CA 94303 USA (e-mail: zhan.shen@sofia.edu).

David Whiteman and Javier Gervas-Arruga are with the Takeda Development Center Americas Inc., Lexington, MA 02142 USA (e-mail: david.whiteman@takeda.com; javier.gervas@takeda.com).

Digital Object Identifier 10.1109/TNSRE.2024.3416159

***Index Terms*— Cerebral Palsy, gross motor function, computer vision, metric learning, transfer learning, spatial-temporal graph convolutional network.**

I. Introduction

THE assessment of motor function in individuals affected by a range of diseases, particularly those with neurological origins, plays a pivotal role in modern healthcare. Gross and fine motor skills, which encompass daily activities related to mobility such as walking, balance, postural control, and to arm/hand control such as reaching to and manipulating objects, are integral to an individual's independence and quality of life. Various neurological disorders, including conditions like Cerebral Palsy (CP), Metachromatic Leukodystrophy, Stroke, and Parkinson's, may significantly impair an individual's motor control and coordination abilities. The importance of assessing motor abilities and capabilities via standardized clinical assessments spanning the World Health Organization (WHO) International Classification of Functioning, Disability, and Health (ICF) cannot be overstated. The ICF, developed by the WHO, provides a unified framework for describing health and health-related domains. These domains are categorized into two lists: 1) body functions and structures, such as muscle power and brain function, and 2) activities and participation, like task execution and involvement in life situations. As a research tool, the ICF helps measure patient outcomes and categorize quality of life or environmental factors affecting performance.

The Gross Motor Function Classification System (GMFCS) is a widely used functional assessment tool for children with CP, comprising five levels that range from those who can independently walk or run on all surfaces (level I) to those with severely limited mobility requiring assistive devices (level V) [2], as summarized in Table I. Typically, GMFCS demonstrates good performance in terms of both inter-rater reliability, which concerns agreements in assessments between different evaluators, and intra-rater reliability, which pertains to consistency in evaluations by the same assessor over time [3], [4]. However, being a qualitative and rudimentary nominal scale, the GMFCS presents challenges when used by non-professionals. Previous research has shown that the inter-rater agreement between parents and therapists is much lower than the agreement between two therapists, and parents tend to assign a higher GMFCS level to the patients [5],

TABLE I
DESCRIPTION OF THE 5 GMFCS LEVELS

Level	Description
I	- Can walk indoors and outdoors and climb stairs without using their hands for support. - Can run and jump. - Has decreased speed, balance, and coordination.
II	- Can walk indoors and outdoors and climb stairs using a railing. - Experiences difficulty with uneven surfaces, inclines, or while in crowds. - Can minimally run or jump.
III	- Walks with assistive mobility devices indoors and outdoors on level surfaces. - May be able to climb stairs using a railing. - May propel a manual wheelchair; may require assistance for long distances or uneven surfaces.
IV	- Walking ability is severely limited, even with assistive devices. - Uses a wheelchair most of the time and may propel their own power wheelchair. - May participate in standing transfers.
V	- Has physical impairments that restrict voluntary movement control and the ability to maintain head and neck position against gravity. - Experiences impairment in all areas of motor function. - Can't sit or stand independently, even with adaptive equipment. - Can't independently walk, though may be able to use powered mobility devices.

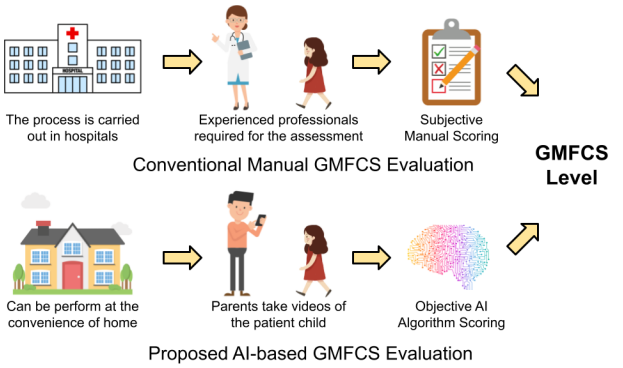


Fig. 1. The AI-based GMFCS assessment is quick, cost-effective, and convenient, enabling detailed patient monitoring.

[6]. In some studies, the inter-rater variability can be as high as $\kappa_{lw} = 0.64$ between a clinical physiotherapist and a researcher, and $\kappa_{lw} = 0.57$ between a clinical physiotherapist and a parent/guardian [7]. Consequently, GMFCS assessments usually require clinic visits, where clinical evaluators observe and categorize a child's movement abilities by instructing them to perform various physical exercises. The assessment session for the GMFCS and other scales can be time-consuming and must be repeated regularly to assess the effectiveness of interventions, placing a considerable time burden on families or caregivers throughout the course of treatment.

In this study, we introduce an innovative approach that uses machine learning and computer vision to assess motor function from monocular videos of patients walking or running, recorded with consumer-level devices. For example, parents of children with Cerebral Palsy can easily capture a video of their child's gait, and our program can promptly evaluate GMFCS levels on their personal devices. Compared to conventional methods, our approach offers significant advantages: it allows evaluations outside hospital settings for greater convenience, reduces inter-rater and intra-rater variability for consistent assessments, and enables continuous and fine-grained monitoring—features not possible with traditional methods.

Computer vision techniques have made significant advancements in recent years, leading to growing interest in their application for patient movement analysis. The majority of these initial studies primarily concentrate on early prediction of Cerebral Palsy. These investigations have harnessed the power of machine learning techniques, alongside the incorporation of hand-crafted features [8], [9], [10], [11], with limited attention given to GMFCS estimation. A notable exception is the work by Kidziński et al. [12]. Their approach involved the use of a simple end-to-end 1D convolutional neural network, which operated on time-series data derived from expert-defined keypoints and hand-crafted features. Our proposed method represents a significant step forward in this direction. We adopt a more advanced network architecture, Spatial-Temporal Graph Convolutional Networks (STGCN), to encode human movements. Furthermore, our approach employs a data-driven method that is not reliant on expert-defined features, thereby effectively addressing the limitations associated with traditional feature-based engineering.

Healthcare data limitations, such as data scarcity, privacy issues, and quality challenges, hinder the full potential of machine learning in the sector, often leading to overfitted models that perform poorly in testing. To address this, we propose a metric learning approach using triplet loss and consistency loss to develop a robust movement encoder. This enhanced encoder is crucial for our GMFCS level and confidence estimation strategies. Our results show that this method significantly outperforms end-to-end models and provides confidence estimates, addressing uncertainty in healthcare data analysis [13].

We can summarize our innovative contributions as follows:

- We introduce the use of STGCN to encode sequences of human pose data for GMFCS assessment, and perform a head-to-head comparison with the current state-of-the-art approach.
- We introduce a metric learning approach that combines triplet loss with self-supervised learning to train a robust STGCN movement encoder, based on which we further propose methods for retrieval-based GMFCS level and confidence estimation.
- We release the code for academic non-commercial use.[1]

II. METHODS

A. Overview

Our method, illustrated in Fig. 2, comprises two stages: first we extract a sequence of poses from the input video, and

[1] https://github.com/the77lab/gmfcs_stgcn

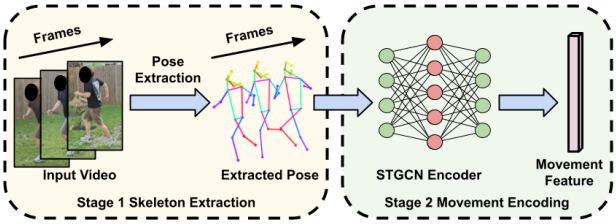


Fig. 2. Overview of our proposed method.

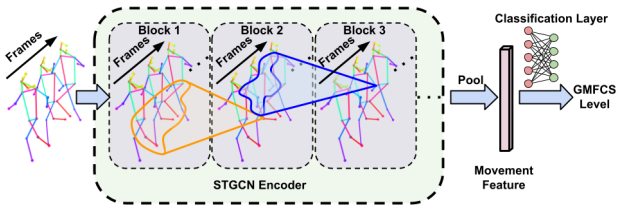


Fig. 3. The architecture of end-to-end GMFCS assessment model.

then we employ this time-series of poses as the input for the second-stage neural network to encode the movement.

To obtain poses from the input video, we can employ established off-the-shelf methods known for their effectiveness. For example, we can use OpenPose [14], which is a bottom-up approach where all human joints present in the image are first detected, and then the detected joints are connected to form skeletons for each individual within the scene. The human skeleton data in the open dataset that we used in this research is extracted with OpenPose.

After extracting the poses, we have a collection of keypoints denoted as $I \in \mathbb{R}^{T\times V\times 3}$, where T signifies the total number of timestamps, and V represents the count of keypoints within a human pose. Each keypoint is characterized by a 3-dimensional vector that encompasses the pixel coordinates for x and y, along with the confidence score for the keypoint's detection. The central innovation in this paper is focused on the second stage, which we introduce in the remaining part of this section.

B. Spatial-Temporal Graph Convolutional Networks

Since its first adoption in Human Action Recognition (HAR) [15], STGCN has attracted significant research interest in the HAR community. STGCN, along with its variants, combines principles from graph theory and convolutional neural networks (CNNs), where it extracts spatial features based on graph topology with graph convolutional networks (GCN), modeling the spatial relationship between different joints, and temporal information with temporal convolutional networks (TCN) which encodes the movement features. They are particularly well-suited for tasks like action recognition in videos, where the spatial position of body joints or objects and the temporal sequence of actions needs to be analyzed. Note that we also adopt a residual connection, except for the first block, with a possible convolutional operation to change the channel width when the input and output feature dimensions are different. The matrix I that represents the sequence of keypoints is directly used as the input of the first STGCN block:

$$X_1 = TCN_0(GCN_0(I)) \tag{1}$$

and for an intermediate block with an input $X_i \in \mathbb{R}^{T_i\times V\times C_i}$, the operation in each STGCN block can be represented as:

$$X_{i+1} = TCN_i(GCN_i(X_i)) + RES_i(X_i) \tag{2}$$

where $i \in [1, N-1]$, $X_{i+1} \in \mathbb{R}^{T_{i+1}\times V\times C_{i+1}}$, and RES is a 1×1 convolutional operation when $C_{i+1} \neq C_i$. For certain blocks we can apply a maxpooling with a size of 2 and a stride of 2 along the time dimension, leading to T_{i+1} being halved compared to T_i.

From different variations of STGCN, we choose to use a GCN with trainable adjacent matrix, and a multi-branch TCN as introduced in STGCN++ [16]. Following this pattern, we alternately extract spatial and temporal features. Suppose we have B blocks, we can get an intermediate feature representation $X_B \in \mathbb{R}^{T_B\times V\times C_B}$, where T_B is the number of timestamps after temporal convolution, and C_B is the dimension of latent feature. We further apply average pooling on both temporal and spatial dimensions, and we can get a latent representation of the movement in the video as $E \in \mathbb{R}^{C_B}$.

C. End-to-End Model

The most straightforward approach for GMFCS level assessment involves employing an end-to-end architecture, as we proposed in our conference paper [1], which serves as a head-to-head comparison of motion encoder to the previous state-of-the-art [12]. In this architecture, linear layers are utilized to map the latent feature to multiple neurons, each corresponding to a GMFCS level, as depicted in Fig. 3. A softmax activation function is applied to the output layer, and the GMFCS level with the highest probability is selected as the predicted level. In our proposed model, we employ two linear layers to map the latent feature to the output layer, which consists of 4 neurons representing the 4 GMFCS Level (GMFCS level V is self evident and excluded from this work). This mapping process can be expressed as follows:

$$y_{out} = softmax(W_{l2}(ReLU(W_{l1}E + b_{l1}) + b_{l2}) \tag{3}$$

where W_{l1} and W_{l2} represent the weight matrices for the first and second linear layers, respectively. b_{l1} and b_{l2} are the bias terms associated with the linear layers. And finally, the predicted GMFCS level is the argmax of y_{out}. The end-to-end model can be trained with cross-entropy loss.

Transfer learning has been shown to be particularly effective in scenarios where the training dataset is limited in size. Given the limited availability of medical data, we employ transfer learning by leveraging a pre-trained STGCN model from an action recognition dataset known as "NTU RGB+D 120" [17]. We adapt it and discard the original classification layers from the pre-trained model, which were designed for action recognition, and substitute them with our own classification module.

D. Metric Learning With Triplet Loss and Self-Supervised Training

In healthcare applications, limited sample sizes, especially those professionally labeled, are common. This often leads to overfitting and reduced testing performance, as demonstrated

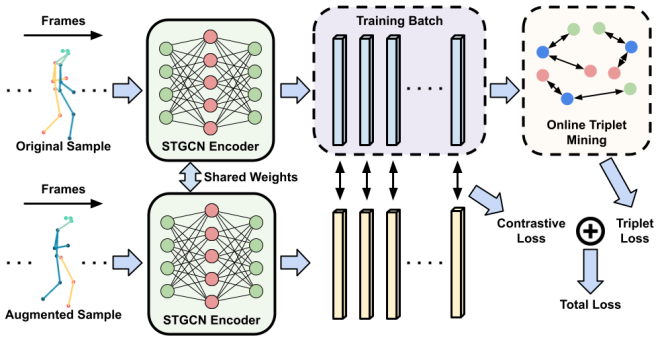


Fig. 4. The workflow of our proposed method.

in Section III. To improve performance with such limitations, we further improve the basic end-to-end model by employing Metric Learning for training the STGCN encoder and using an instance-retrieval based classification method. The idea behind metric learning is to ensure that embedding of samples from different GMFCS levels are far apart in the latent space, while embedding of samples from the same GMFCS level are closer. This encourages the extraction of distinct features for clear movement identification and classification. Our metric learning framework is illustrated in Fig. 4.

We use a combination of triplet loss and consistency loss to train the network, with a weight of 1:1 following:

$$\mathcal{L} = \mathcal{L}_{triplet} + \mathcal{L}_{consistency} \tag{4}$$

Triplet loss serves to compel the network to extract distinct features for samples from different GMFCS Levels while ensuring similar features for samples from the same level. To implement triplet loss, we create triplets comprising an anchor sample, a positive sample (another sample with the same GMFCS level), and a negative sample (with a different GMFCS level). These triplets are encoded using the STGCN backbone as E_a, E_p, and E_n. The triplet loss can be expressed as follows:

$$\mathcal{L}_{triplet} = max(0, \|E_a - E_p\|_2 - \|E_a - E_n\|_2 + m) \tag{5}$$

Here, m represents the margin we aim to establish between the distances of positive pairs and negative pairs within a specific triplet. In other words, for a given triplet, if the distance between the negative pairs is not greater than the distance between the positive pairs by a margin of m, this pair is penalized and contributes to the update of the STGCN encoder weights through back-propagation. It's important to note that the margin is important for effective network training. Without it, the encoder might simply produce identical embedding for all input samples to meet the constraints.

On the other hand, observing movement introduces aleatory uncertainty [18], stemming from camera position, orientation, occasional occlusion, key point mis-detection, etc. To enhance the robustness of the feature encoder, we adopt a loss function named Consistency Loss, which focuses on minimizing the feature distance between a pose sequence and its augmented version.

For instance, if we have an original sample I_{ori}, we apply a random data augmentation to produce I_{aug}. We then feed both samples into the network to obtain their latent representations, E_{ori} and E_{aug}. The adopted consistency loss is expressed as follows:

$$\mathcal{L}_{consistency} = max(0, \|E_{ori} - E_{aug}\|_2 - \epsilon) \tag{6}$$

where ϵ is an empirical value to allow a feature distance threshold between the original sample and its augmented version. Here we use an ϵ value of 0.1.

We incorporate the following three data augmentation techniques:

- A random Shear Transformation, which simulates variations in camera position and orientation.
- A Mirror Transformation, involving the swapping of keypoints between the left and right sides.
- A Masking operation, randomly concealing one of the keypoints from the upper body and its neighboring keypoints, simulating the scenario of missing body part detection.

Each of the manipulations is applied with a 50% chance. It's important to note that the augmented sample preserves the original sample movement. Ideally, it would be encoded into the exact same embedding as the original sample. However this can be too demanding for the encoder because the input is different. Therefore, we introduce a feature distance threshold to relax slightly this constraint and promoting a more stable training, as depicted in Equation 6.

E. GMFCS Level and Confidence Estimation

In our GMFCS level estimation, we employ a retrieval-based approach and estimate confidence based on the retrieved embedding and their corresponding labels. During testing, the training samples serve as the support set. We calculate the embedding for each training sample and store the mapping between embeddings and their labels.

For a given test video, we segment it into samples. With each testing sample, we compute a latent embedding using the trained STGCN encoder. Subsequently, we retrieve the nearest k embedding from the support set, denoted as $N = \{n_i | i = 1, 2, \ldots, k\}$, along with their corresponding labels, represented as $L = \{l_i | i = 1, 2, \ldots, k\}$, from the support set. We then calculate the distances between the testing sample and each of the embedding in N, resulting in $D = \{d_i | i = 1, 2, \ldots, k\}$.

Using these distances, we compute the probability of the testing sample belonging to different GMFCS levels, which further leads to an overall GMFCS level and a confidence estimation for the entire video input. The complete procedure is outlined in Algorithm 1.

There are a few cases where we aim to assign low confidence during the final voting process. The first category comprises samples with Out-of-Distribution uncertainty, meaning the movement observed in the testing sample has never been encountered in the training set. This is particularly likely when the training set is small, and it's indicated by a relatively large distance between the test embedding and the nearest embedding in the support set. The second category includes testing samples that lack sufficient information to clearly determine their GMFCS level. For instance, samples

Algorithm 1 GMFCS Level and Confidence Estimation

```
test_embeddings ← ENCODER(test_samples)
sample_results ← []
for all test_embedding in test_embeddings do
    N ← GETSUPPORTSAMPLES(test_embedding)
    L ← GETSUPPORTLABELS(N)
    D ← GETSUPPORTDISTANCES(N)
    P ← GETLABELPROBS(D, L)
    Append P to sample_results
end for
label, confidence ← GETLABELANDCONFIDENCE(sample_results)
```

containing the movement of a patient slowly turning around may pertain to either Level I or Level II patients, and a single sample does not provide enough information for a definitive decision. In terms of the latent feature space, the closest embedding may belong to both Level I and Level II, both with small distances, prompting us to assign low confidence to these types of samples. Additionally, a patient's gross motor function ability might lie between two GMFCS levels, making it challenging to provide a confident estimation of the patient's specific GMFCS Level.

In our proposed approach, as described in Algorithm 1, we employ the following equations to calculate the probability that a sample falls into each GMFCS level, represented as $P = [p_1, p_2, p_3, p_4]$:

$$p_j = \begin{cases} \dfrac{1}{mean(d_{j,i})}, & \exists i, n_{j,i} \in N \\ 0, & otherwise \end{cases} \tag{7}$$

where $mean(d_{j,i})$ is the average distance between the testing embedding and the embedding in N that belongs to GMFCS Level j. After that we perform a softmax operation on P to get the probability:

$$P \Leftarrow softmax(P) \tag{8}$$

After we get the probability vector P for each testing sample, we perform the final GMFCS level and confidence estimation following Eq. 9 and Eq. 10. Suppose we have a total of M testing samples, we have:

$$GMFCS = \underset{j}{\text{argmax}} \sum_{m=1}^{M} p_{j,m} \tag{9}$$

where $p_{j,m}$ represents the confidence of label j for the m^{th} testing sample. Furthermore, the confidence of the video can be calculated as:

$$Confidence = \frac{\sum_{m=1}^{M} p_{GMFCS,m}}{\sum_{j=1}^{4} \sum_{m=1}^{M} p_{j,m}} \tag{10}$$

Note that the confidence here is a manually defined metric, and it does not directly reflect the probability of the GMFCS estimation being correct, e.g., a confidence of 0.75 does not indicate that there is a 75% chance of the current video being correctly classified.

We have the option to reject the GMFCS estimation if the confidence is low. In such cases, the video can be forwarded to experts for manual labeling, contributing to continuous learning. By retaining only the estimations with high confidence, we enhance the accuracy of GMFCS level estimation, as illustrated in Section III.

III. Evaluation

We initiate the evaluation by assessing the end-to-end model, which was proposed in our original conference paper [1]. This evaluation mainly compares the STGCN encoder to the 1DCNN proposed in the previous state-of-the-art [12]. We further delve into ablation studies concerning transfer learning policies, and explore the correlation between training set size and the performance of the end-to-end model, setting the stage for subsequent experiments on metric learning and confidence estimations. Additionally, we discuss instances of error cases encountered during the evaluations. Finally, we evaluate the on-device inference speed of all the proposed methods.

A. Dataset Preparation

We use a publicly available dataset from Kidzinski et. al. [12] for the evaluations. This dataset contains videos from CP youngsters collected in a clinical setting with their GMFCS level assessed by health care professionals (ground truth). Average age of the youngsters is 11 $y.o.$ ($s.d.$ = 5.9), with average height of 133cm ($s.d.$ = 22), and weight of 34kg ($s.d.$ = 17). The original paper lacks some details on how to reproduce the exact training, validation and testing split. As we cannot get the same dataset split running their provided code, we use our own protocol for pre-processing.

We use data with GMFCS levels I to IV, because children at level V cannot move by themselves. We check all the skeleton videos and manually remove 85 videos that contain more than one person, resulting in 1,450 videos from 861 patients. We split the dataset into training, validation and testing. We use stratified sampling and sample each GMFCS level separately. For each GMFCS score, we split the dataset using the patient's ID with ratio of 7:1:2, as we want a patient, labeled with a specific level, to appear in only one of the training, validation, and testing dataset. The detailed ground truth GMFCS level distribution of the dataset is shown in Fig. 5.

B. Method Implementation

We implemented our GMFCS assessment model and metric learning approach using PyTorch, integrating code and a pre-trained STGCN model from Pyskl [16]. The training involves initially keeping the STGCN backbone fixed for 3 epochs while training the classification layers, then fine-tuning the last 2 STGCN blocks. We used the Adam optimizer with a learning rate of 1e-4 and a weight decay of 5e-5, running the model for 10 epochs with a batch size of 128. Model selection was based on the highest validation accuracy, using cross-entropy loss.

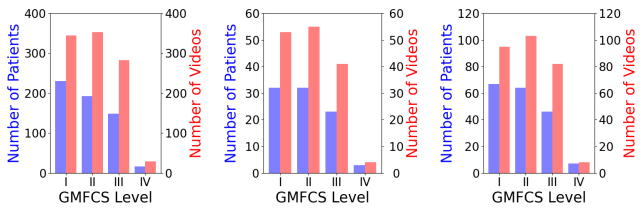


Fig. 5. Number of patients and videos in the dataset for training, validation, and testing (left to right panel).

In the context of metric learning, we incorporate the open-source library pytorch-metric-learning [19] to execute the triplet loss calculation and online triplet mining. Online triplet mining dynamically selects triplets during the training process, generating triplets within the current batch. This method avoids the need for precomputed triplets, leading to better adaptability, efficiency, and potentially improved model performance compared to fixed triplets used in offline mining.

Our triplet loss function employs a margin of $m = 0.6$, and we utilize $\epsilon = 0.1$ in the consistency loss. The model undergoes training for 40 epochs, with a batch size set at 128 and a learning rate of 1e-4. While we unfreeze the final 2 STGCN blocks for fine-tuning in the end-to-end model, we unfreeze the last 6 STGCN blocks from epoch 0 in the metric learning. For model selection, we adopt a straightforward approach, which entails using the label of the support embedding closest to the validation sample embedding as an estimate of the validation sample's label. The best model is chosen based on its performance in terms of validation accuracy. In the retrieval-based method for GMFCS assessment, we retrieve $k = 20$ nearest neighbors for each testing sample.

C. End-to-End Model Evaluation

We assess the efficacy of our proposed end-to-end approach on the testing dataset, wherein each sample undergoes classification, and the final outcome is determined through a majority voting mechanism. To gauge the performance of our method, we conduct a comparative analysis against the previous state-of-the-art approach [12]. The previous method involves segregating the displacement of 8 joints and an additional 8 hand-crafted features into distinct channels within the time-series data. It then employs a 1D Convolutional Neural Network for subsequent temporal feature extraction. A key distinction between our approach and the prior method lies in our avoidance of manual feature selection; instead, we leverage spatial constraints, in the form of graph topology, during training to amalgamate information from various joints.

For the sake of a fair and objective comparison to the baseline [12], we execute their official code directly on our dataset split, making use of their provided pre-processing, training, and testing pipelines. Our results with their method surpass the figures reported in their original paper, which had indicated an accuracy of 66%. To mitigate the impact of randomness inherent in network training, we run each method 5 times. The prior state-of-the-art method manages to attain an accuracy of 71.61% ($s.d.$ 0.76%), whereas our method demonstrates better performance with an accuracy of

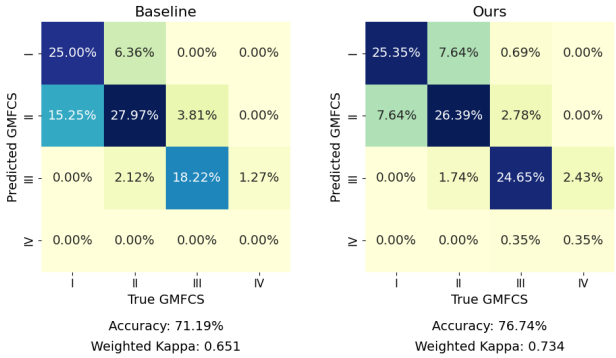


Fig. 6. Our proposed method outperforms the previous method in terms of accuracy and linear weighted Cohen's Kappa, which is used to measure the agreement of two voters, i.e., clinician and AI algorithm.

76.60% ($s.d.$ 0.35%) and an average $\kappa_{lw} = 0.733$. To facilitate a more comprehensive analysis, we select one model from each approach and compare their outcomes in Figure 6.

As can be seen in the results, our proposed approach has an accuracy of around 5% higher than the previous approach. According to the confusion matrix, the errors mostly happen between Level I and Level II. This is because GMFCS Level I and Level II are inherently similar, and it's very challenging for machine learning methods to learn subtle differences. Furthermore, the ground truth labels provided by healthcare professionals could be sub-optimal, as these two levels could be confusing to human raters as well. As for the two methods compared here, our proposed approach correctly classifies 75.3% of the Level I and Level II samples, while the baseline approach has an accuracy of 69.1%. We believe that this is due to the much stronger representation ability of our proposed model, which captures more subtle features in these two levels. As a result, our method could possibly perform even better if the quality and quantity of the training dataset further improve. Also, we can see that both models struggle to correctly classify Level IV samples, as we do not have enough Level IV training data. Also, our method has a linear weighted Kappa of 0.734, a significant improvement over the previous method with a Kappa of 0.651, demonstrating a substantial agreement with the ground truth labels. This proves that our proposed method has great potential for accurate GMFCS Level estimation.

D. Ablation Study on Transfer Learning of End-to-End Model

For a more comprehensive understanding of our proposed method, we conducted an ablation study, where we compare the following 4 methods: (1) "Ours" refers to our proposed transfer learning policy introduced in Section II, where we only fine-tune the last 2 STGCN blocks of the pre-trained model; (2) "Fixed", where the weights of the backbone STGCN remained fixed after loading the pre-trained model; (3) "All", where all the blocks of the backbone STGCN were fully unfrozen for fine-tuning; (4) "No-Pre", where the weights of the backbone STGCN were trained from scratch using the CP dataset. The results are presented in Fig. 7.

Notably, when we kept the STGCN weights fixed to the pre-trained values, the accuracy significantly deteriorated. This

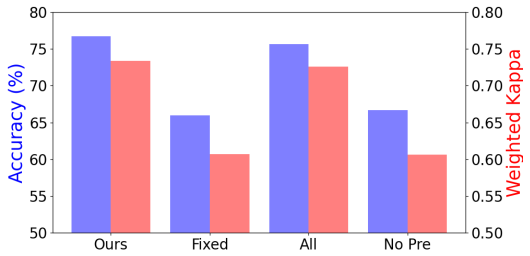


Fig. 7. Ablation study of the transfer learning strategy.

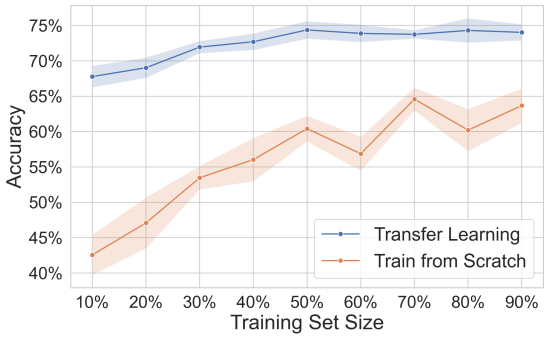


Fig. 8. Impact of training set size on end-to-end model accuracy.

decline may be attributed to the domain differences between action recognition and GMFCS scoring. Conversely, when all STGCN blocks were made trainable, the accuracy was only marginally inferior to our proposed training method, indicating the robustness of our approach to the degree of STGCN block fine-tuning. However, if we didn't load the pre-trained weights and trained the model from scratch, the performance suffered significantly, underscoring the pivotal role of transfer learning in our case.

E. Impact of Training Set Size on End-to-End Model Accuracy

As demonstrated in the previous section, transfer learning plays a critical role in mitigating overfitting and improving model performance during testing. In this section, we delve deeper into evaluating the influence of training data size on the testing accuracy of the end-to-end model. We conduct a stratified sampling of our training set, selecting 10%, 20%, ..., 90% of the original training data. We compare the testing accuracy of training from scratch and using transfer learning. To ensure robust results, each experiment with a specific training set size is repeated 5 times, both for sampling and training. The findings are summarized in Fig. 8.

As evident in the outcomes, when considering both transfer learning and training from the ground up, the accuracy demonstrates an upward trajectory as the volume of training data increases. Nevertheless, the disparity in accuracy between these two approaches widens notably when dealing with a smaller quantity of training data. Specifically, when working with only 10% of the original training data, employing transfer learning still yields testing accuracy in the range of 65% to 70%, whereas starting from scratch results in a significantly lower accuracy range of approximately 40% to 45%, akin to

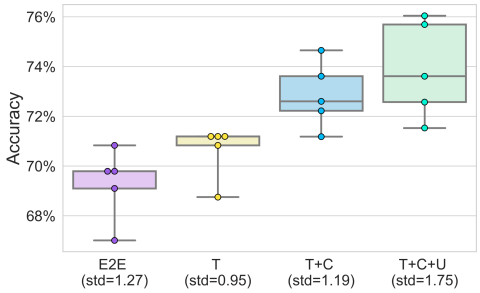


Fig. 9. With 10% of the original training dataset size, metric learning methods are able to achieve significant better performance than the end-to-end model.

random guessing. These results further underscore the pivotal role of transfer learning in medical applications, particularly in scenarios where the availability of labeled data is limited.

F. Metric Learning Evaluation

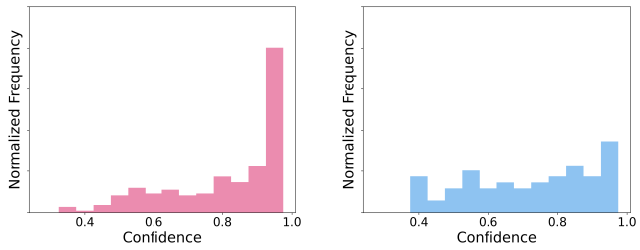
We further evaluate the proposed metric learning and retrieval-based GMFCS assessment approach, as a possible solution to the impact of limited amount of training data. We first conduct an experiment, where we sampled 10% of the original training set with stratified sampling, and repeat the training of each method 5 times. When it comes to medical data, it could also be highly possible that we would have a larger amount of unlabeled data, and we also take this into consideration. We compare the following 4 methods here:

1) End-to-end model. (E2E)
2) Metric Learning with Triplet Loss Only. (T)
3) Metric Learning with Triplet Loss and Consistency Loss. (T+C)
4) Metric Learning with Triplet Loss and Consistency Loss. Using the remaining 90% of training set data as unlabeled data. (T+C+U)

For Method 4), when calculating consistency loss, we keep the same batch size as in 3), while randomly sample the batch from the whole training set instead of from the 10% as in triplet loss. The results of this experiment is shown in Fig. 9. Note that to have a fair comparison, for T, T+C and T+C+U methods, we adopt a simple labeling approach, where for each testing sample, we assign a GMFCS level based on majority voting of 5 nearest neighbors, and the final video label is estimated using the majority voting of sample labels.

As observed in the figure, training the end-to-end model directly on a training set with only 10% of the original size results in suboptimal performance, with an average accuracy below 70%. However, the inclusion of metric learning methods significantly improves the results. We conducted Welch's t-test [20] on pairs of the results, as summarized in TABLE II.

The t-test analysis reveals that models trained with the triplet loss exhibit a relatively high confidence in outperforming the end-to-end models. Importantly, we have strong confidence (with $p < 0.01$) that models trained with the triplet + consistency loss, whether utilizing the remaining unlabeled data or not, consistently surpass end-to-end models and models trained with only the triplet loss. The inclusion of the remaining 90% unlabeled data shows potential for further enhancing model



(a) Correct Estimations (b) Incorrect Estimations

Fig. 10. Confidence distribution for correct and incorrect estimations.

TABLE II
T-TEST RESULTS FOR METRIC LEARNING EVALUATION

Comparison	P-Value
E2E vs. T	0.068334
E2E vs. T+C	0.001800
E2E vs. T+C+U	0.001734
T vs. T+C	0.010154
T vs. T+C+U	0.008095
T+C vs. T+C+U	0.179590

performance. However, given the p-value's limited confidence and the real-world constraints regarding the availability of unlabeled data, this aspect warrants future exploration.

G. Retrieval-Based Classification and Confidence Estimation Evaluation

We further conduct the evaluation of the Retrieval-based Classification and Confidence Estimation method, employing a model trained with only 10% of the original training data set size using triplet loss and consistency loss. Following the approach outlined in Section II, we compute GMFCS label predictions and their corresponding confidence estimates for each testing video. The confidence distribution for correct predictions and incorrect predictions is illustrated in Fig. 10.

As depicted in the illustration, erroneous predictions typically exhibit reduced confidence and seem to follow a more evenly spread distribution when contrasted with accurate predictions. On the other hand, the majority of accurate predictions boast notably high confidence levels. These disparities in distribution open up avenues for establishing specific confidence thresholds that could potentially enhance the accuracy of GMFCS assessments.

In a follow-up experiment, we vary the confidence thresholds and assess how the accuracy and the proportion of confident estimations change in response to different threshold values. The outcomes of this experiment are presented in Fig. 11.

The results illustrate a clear and notable trend. When the confidence threshold is set below 0.4, all GMFCS Level estimations are deemed confident, resulting in an accuracy of 76.04%, which is already quite good, as it closely approaches the performance of the end-to-end model trained with a full-sized training set. As the confidence threshold is increased to accept fewer GMFCS Level predictions, accuracy rises. It reaches 88% when the confidence threshold is set at 0.95, where approximately 34.72% of the testing video GMFCS estimations are considered confident. In practical applications,

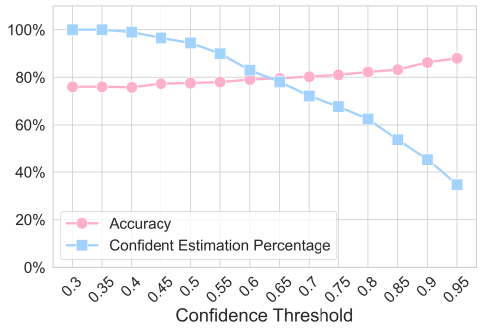


Fig. 11. The change of accuracy and percentage of confident samples with different confidence thresholds.

these confidence thresholds can be adjusted to meet specific requirements.

H. Visualization and Error Cases Study

To better grasp the assignment of confidence scores to testing videos, we visualize the support and testing sample embeddings in a lower-dimensional latent space using t-SNE [21], a commonly employed dimensionality reduction technique for data visualization. Here the t-SNE reduces the embedding dimension to 2 for visual representation, available in Fig. 12.

The visualization provides valuable insights. For GMFCS estimations with high confidence, the embeddings of the testing samples are clearly located within one of the distinct clusters representing Level I to IV (e.g. Fig. 12 (a)). Conversely, when the testing embedding occupy a space between two clusters or span across different clusters, the final GMFCS estimation is associated with lower confidence. In such cases, correct predictions can still occur, as seen in Fig. 12 (b). However, there is also the possibility of an incorrect prediction, as demonstrated in Fig. 12 (c), where some of the testing samples are within the Level I cluster and others within the Level II cluster. This observation may suggest that the patient's gross motor function falls somewhere between Level I and II.

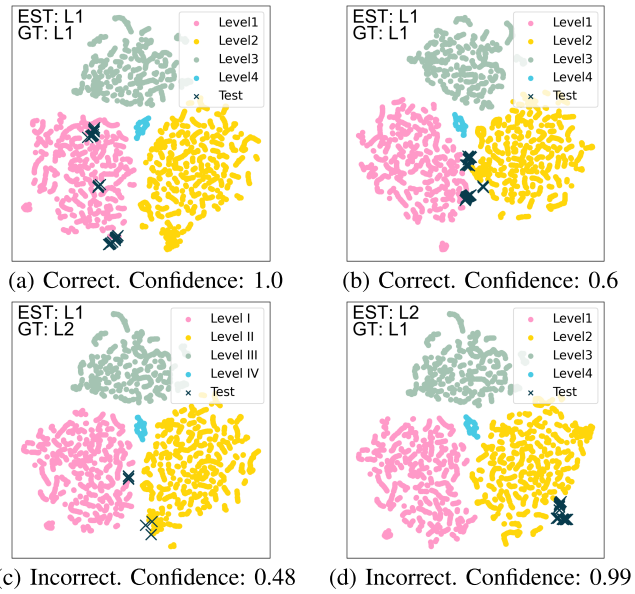
Fig. 12 (d) shows a scenario where we have a high confidence on a wrong GMFCS Level estimation. We argue that this could due to two reasons:

- **Inaccurate Ground Truth Labeling**: The ground truth labels are not always perfectly accurate. As discussed earlier, these labels are provided by healthcare professionals, and there can be inherent intra-rater and inter-rater variability. Consequently, a video labeled as Level II might actually correspond to Level I.
- **Overfitting to Spurious Features**: Overfitting to certain non-representative features is another possibility, especially when dealing with a limited training dataset. This issue may potentially be mitigated through lifelong learning and continuous model refinement.

In real-world scenarios, a pragmatic strategy involves sending videos with low-confidence GMFCS estimations to healthcare professionals for manual labeling and cross-verifying results from confident estimations. This iterative process accumulates valuable data for fine-tuning the encoder, strengthening its robustness and enhancing the GMFCS estimation

TABLE III
RUNNING TIME ON MOBILE DEVICES AND LAPTOPS (MS)

Device	Platform	CPU	GPU	PoseNet	STGCN_E2E	STGCN_B	Sample	Video
Samsung Galaxy ZFlip 5	Android	Snapdragon 8.2	Adreno 740	37.3	106.9	106.7	6.6	0.03
Samsung Galaxy Tab S8+	Android	Snapdragon 8.1	Adreno 730	47.7	121.4	118.2	8.0	0.03
Google Pixel 4a	Android	Snapdragon 730G	Adreno 618	66.7	489.9	490.2	42.3	0.02
Apple iPhone 7 plus	iOS	A10 Fusion	PowerVR 7XT+	87.6	873.2	732.6	18.4	0.03
ASUS ROG Strix	Windows	Core i9-12900H	RTX 3080Ti	8.2	76.7	86.5	3.2	0.006
ASUS Zenbook Pro	Windows	Core i7-12700H	Intel Iris Xe	16.8	81.0	79.4	4.2	0.007



(a) Correct. Confidence: 1.0 (b) Correct. Confidence: 0.6

(c) Incorrect. Confidence: 0.48 (d) Incorrect. Confidence: 0.99

Fig. 12. t-SNE visualization of correct and incorrect GMFCS level estimations with different confidence.

system's accuracy and reliability. Over time, this approach ensures continuous improvement of the system, adapting to a wider range of scenarios and reducing the margin of error.

I. Running Time Evaluation

Given the sensitive nature and privacy of patients' video data, it is ideal to perform all computations on end-user devices, ensuring no visual data is transferred online. To achieve this, the proposed method must be efficient enough to run on various platforms. We use a web app to evaluate the runtime of the proposed method on mobile devices, enabling cross-platform adoption. Notably, although it is a web application, all computations occur on the client-side, with no visual data transferred to a server.

For the demonstration, PoseNet [22] from Tensorflow.js is employed as the pose extractor, running with the WebGL backend on a GPU. The STGCN PyTorch model is converted to an ONNX model and executed with ONNX Runtime Web. Due to some unsupported operators with ONNX Runtime Web's WebGL backend, it is run with the WASM backend, utilizing the CPU. All the calculation afterwards are based on native JavaScript code. The experiment is conducted within a React application running in the Chrome browser. The results are summarized in Table III. The "STGCN_E2E" column reflects the runtime of the end-to-end model, the "STGCN_B" column illustrates the runtime of the STGCN backbone for calculating the latent representation, and the "Sample" column represents the runtime for retrieval of the nearest $k = 20$ neighbors and the calculation of the label probability vector for a single testing sample, with a support set size of 3000. Finally, the "Video" column shows the run time of the final decision making and confidence estimation process, based on 30 testing samples.

Results show that pose extraction achieves around 30 FPS on modern mobile phones like the Samsung Galaxy ZFlip 5 using the onboard GPU, and performs even faster on laptops with high-performance graphics cards. Although the STGCN runs slower on the CPU, it is only called intermittently when enough frames (124) are available, and can run in parallel with pose extraction. Additionally, retrieval-based classification and confidence estimation are quick, even on older devices such as the iPhone 7 Plus (18.43 ms). Thus, the proposed method operates nearly in real-time on mobile devices.

IV. DISCUSSION

The GMFCS is a nominal functional scale used for classification, but it could also be viewed as a regression problem. Classification aims to create distinct features for different classes to minimize confusion, a principle we used in our metric learning approach. However, impairment is a continuous process, and its features should form a continuous pattern in the feature space. Since the ground truth labels are discrete levels, direct regression is impractical as it would cause the network to map samples from each class to a single point, limiting the effectiveness of a regression-based approach. In future work, exploring methods for training the model using a regression approach could be highly beneficial, as it aligns better with the nature of the evaluation problem. This might involve developing finer and more continuous ground truth labels to capture gradual changes in gross motor function. Studies using robot-based kinematic measurements in stroke, such as those by [23], [24], [25], [26], and [27], suggest that a regression approach could yield a significantly larger effect size compared to traditional nominal clinical scales, potentially reducing the patient census needed to test new interventions. Additionally, improving the accuracy of the computer vision scheme, for instance from 71.61% to 76.60%, would also lead to a larger effect size.

Also, in this work we only employed 3 basic skeleton augmentation techniques: Shear, Mirroring and Masking. More advanced augmentation methods, including AdaIN, Gaussian Noise, Gaussian Blur, Rotation, and Skeleton Mix, have been proved effective in recent work [28]. Incorporating these extra enhancements may offer understanding into the model's

behavior when subjected to various data augmentations, and whether they are successful in reducing overfitting of the model. We will further investigate this topic in our future work.

V. Conclusion

In this article, we explore the application of computer vision and AI techniques to estimate GMFCS levels, comparing these AI-driven evaluations with those performed by professional therapists. We employ STGCN-based networks to analyze spatial and temporal features extracted from single-view video data of human motion, offering a robust approach in GMFCS assessment. Our research introduces an end-to-end model that significantly outperforms the current state-of-the-art, achieving an accuracy of 76.60% compared to the prior benchmark of 71.61%. This model demonstrates a high degree of agreement with therapist assessments, evidenced by an average κ_{lw} of 0.733. Additionally, we have developed a novel training strategy for the STGCN encoder using a metric learning approach. This allows for a retrieval-based GMFCS classification system that includes a mechanism for confidence estimation. Notably, this method excels in scenarios with limited training data. By setting a confidence threshold of 0.95, our model attains an accuracy of 88% using only 10% of the training data typically required, which is particularly advantageous for smaller research studies. Furthermore, our proposed solution is capable of running in near real-time on various mobile platforms, enhancing its applicability in diverse clinical and remote settings. This study highlights the significant potential of AI in advancing smart, efficient, and personalized healthcare solutions, particularly in the realm of patient mobility assessment.

References

[1] P. Zhao et al., "Motor function assessment of children with cerebral palsy using monocular video," in *Proc. IEEE 19th Int. Conf. Body Sensor Netw. (BSN)*, Oct. 2023, pp. 1–4.

[2] A. Paulson and J. Vargus-Adams, "Overview of four functional classification systems commonly used in cerebral palsy," *Children*, vol. 4, no. 4, p. 30, Apr. 2017.

[3] D. Piscitelli, S. Vercelli, R. Meroni, G. Zagnoni, and L. Pellicciari, "Reliability of the gross motor function classification system and the manual ability classification system in children with cerebral palsy in Tanzania," *Develop. Neurorehabilitation*, vol. 22, no. 2, pp. 80–86, Feb. 2019.

[4] D. Piscitelli, F. Ferrarello, A. Ugolini, S. Verola, and L. Pellicciari, "Measurement properties of the gross motor function classification system, gross motor function classification system-expanded & revised, manual ability classification system, and communication function classification system in cerebral palsy: A systematic review with meta-analysis," *Develop. Med. Child Neurol.*, vol. 63, no. 11, pp. 1251–1261, 2021.

[5] D. B. R. Silva, L. I. Pfeifer, and C. A. R. Funayama, "Gross motor function classification system expanded & revised (GMFCS E & R): Reliability between therapists and parents in Brazil," *Brazilian J. Phys. Therapy*, vol. 17, no. 5, pp. 458–463, Oct. 2013.

[6] G. Rackauskaite, P. Thorsen, P. V. Uldall, and J. R. Østergaard, "Reliability of GMFCS family report questionnaire," *Disability Rehabil.*, vol. 34, no. 9, pp. 721–724, May 2012.

[7] B. C. McDowell, C. Kerr, and J. Parkes, "Interobserver agreement of the gross motor function classification system in an ambulant population of children with cerebral palsy," *Develop. Med. Child Neurol.*, vol. 49, no. 7, pp. 528–533, Jul. 2007.

[8] K. D. McCay et al., "A pose-based feature fusion and classification framework for the early prediction of cerebral palsy in infants," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 30, pp. 8–19, 2022.

[9] N. Silva et al., "The future of general movement assessment: The role of computer vision and machine learning – a scoping review," *Res. Develop. Disabilities*, vol. 110, Mar. 2021, Art. no. 103854.

[10] D. Sakkos, K. D. Mccay, C. Marcroft, N. D. Embleton, S. Chattopadhyay, and E. S. L. Ho, "Identification of abnormal movements in infants: A deep neural network for body part-based prediction of cerebral palsy," *IEEE Access*, vol. 9, pp. 94281–94292, 2021.

[11] H. Zhang, E. S. L. Ho, and H. P. H. Shum, "CP-AGCN: PyTorch-based attention informed graph convolutional network for identifying infants at risk of cerebral palsy," *Softw. Impacts*, vol. 14, Dec. 2022, Art. no. 100419.

[12] L. Kidzinski, B. Yang, J. L. Hicks, A. Rajagopal, S. L. Delp, and M. H. Schwartz, "Deep neural networks enable quantitative movement analysis using single-camera videos," *Nature Commun.*, vol. 11, no. 1, p. 4054, Aug. 2020.

[13] P. Zhao et al., "Heart rate sensing with a robot mounted mmWave radar," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Sep. 2020, pp. 2812–2818.

[14] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "OpenPose: Realtime multi-person 2D pose estimation using part affinity fields," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 1, pp. 172–186, Jan. 2021.

[15] S. Yan, Y. Xiong, and D. Lin, "Spatial temporal graph convolutional networks for skeleton-based action recognition," in *Proc. AAAI Conf. Artif. Intell.*, 2018, vol. 32, no. 1, pp. 7444–7452.

[16] H. Duan, J. Wang, K. Chen, and D. Lin, "PYSKL: Towards good practices for skeleton action recognition," in *Proc. ACM Int. Conf. Multimedia (ACMMM)*, 2022, pp. 7351–7354.

[17] J. Liu, A. Shahroudy, M. Perez, G. Wang, L.-Y. Duan, and A. C. Kot, "NTU RGB+D 120: A large-scale benchmark for 3D human activity understanding," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 10, pp. 2684–2701, Oct. 2020.

[18] A. Kendall and Y. Gal, "What uncertainties do we need in Bayesian deep learning for computer vision?" in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.

[19] K. Musgrave, S. J. Belongie, and S.-N. Lim, "PyTorch metric learning," 2008, *arXiv:2008.09164*.

[20] B. L. Welch, "The generalization of 'student's' problem when several different population varlances are involved," *Biometrika*, vol. 34, nos. 1–2, pp. 28–35, 1947.

[21] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.

[22] D. Oved, I. Alvarado, and A. Gallo. (2018). *Real-Time Human Pose Estimation in the Browser With Tensorflow.js*. Accessed: Apr. 29, 2023. [Online]. Available: https://blog.tensorflow.org/2018/05/real-time-human-pose-estimation-in.html

[23] C. Bosecker, L. Dipietro, B. Volpe, and H. I. Krebs, "Kinematic robot-based evaluation scales and clinical counterparts to measure upper limb motor performance in patients with chronic stroke," *Neurorehabilitation Neural Repair*, vol. 24, no. 1, pp. 62–69, Jan. 2010.

[24] H. I. Krebs et al., "Robotic measurement of arm movements after stroke establishes biomarkers of motor recovery," *Stroke*, vol. 45, no. 1, pp. 200–204, Jan. 2014.

[25] D. K. Agrafiotis et al., "Accurate prediction of clinical stroke scales and improved biomarkers of motor impairment from robotic measurements," *PLoS ONE*, vol. 16, no. 1, Jan. 2021, Art. no. e0245874.

[26] C. B. Moretti et al., "Robotic kinematic measures of the arm in chronic stroke: Part 2—Strong correlation with clinical outcome measures," *Bioelectron. Med.*, vol. 7, no. 1, pp. 1–13, Dec. 2021.

[27] S. M. Mostafavi, S. Scott, S. Dukelow, and P. Mousavi, "Reduction of assessment time for stroke-related impairments using robotic evaluation," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 25, no. 7, pp. 945–955, Jul. 2017.

[28] L. Lin, J. Zhang, and J. Liu, "Actionlet-dependent contrastive learning for unsupervised skeleton-based action recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Vancouver, BC, Canada, Jun. 2023, pp. 2363–2372.