Clustering of movement profiles of motor-impaired peoplefrom 2D image streams

Mário André Esteves Macedo

***Abstract*—Ageing, motor disorders and accidents are some possible causes that lead people to require help from caregivers, being impossible for patients to have autonomy. Assistive robots have been developed to help them in tasks such as feeding. However, as each patient is a different case, so it is imperative to classify them and adapt the technology to better serve their needs.**

The aim of this work was a methodology which focused on classifying different subjects based on the manifestation of their disabilities in a specific setup. This would enable a better adaptation of robotic systems to adapt to each patient and consequently improve their performance. To achieve this result, a data set was created with recordings of people with and without physical limitations, while performing given tasks.

Data from specific points was acquired from these videos using a facial landmark detector. Nevertheless, this data was incomplete, and so a matrix completion procedure with a Structure from Motion framework was used. Afterwards, the trajectories of the subjects were encoded onto two different features. These features were then used to build a classifier, based on the Bag of Words model, aiming to discern the different types of subjects depending on their performance.

During the clustering phase, it was possible to identify specific patterns associated to specific type of subjects. The results obtained were quite promising as they proved that the feasibility this study.

***Keywords*—Movement Classification, Matrix completion, Structure from Motion, Trajectory Feature, Bag of Words**

I. Introduction

Nowadays, the number of people with disabilities is growing. One of the main reasons behind this fact, relies on the ageing of the population.

Nevertheless, the percentage of youngsters with severe disabilities still reaches 0.7% [1]. A large part of this young population is affected by Cerebral palsy, which is associated with damage to parts of the brain that control movement, coordination, balance and posture, which impedes the normal movement [2]. However, people are affected differently by this disorder, either in the location affected, or the degree of the control over that region. Therefore, classifying these people is a complex task, which still does not have a generalised result.

Currently, there is the Gross Motor Function Classification System, this system aims to provide a standardised measurement of severity of the disorder. This system consist on a set of levels, which are assigned based on how the subjects perform in a set of tasks [3]. One big problem of such classification relies on the subjectivity of such analysis. So, under the same circumstances, different people might classify the same person differently.

With the goal of attaining objectivity, in this thesis it was developed an automatic classifier which aims to differentiate motor impaired subjects based on their physical limitations. This classifier does not need any type of subjective information manually provided. It simply analyses the recordings of the person under analysis while they try to do some specific movements.

The scenario behind this classification is related with assistive robotics, more precisely, using a robotic arm to help feed people with this type of disorder. Currently, some robotic arms have been developed with the objective of feeding people with with upper arm disabilities. However, most of them are not able to adapt the end-effector trajectory to each user. This poses a problem since depending on the disorder of the subject we might expect different patterns from the robotic arm.

In order to help solving this problem, the proposed classifier would be able to classify the person which was using the robotic arm, and based on this information it would be able to adapt its movement to the specific user.

II. State of the art

A. Human motion recognition

Currently, the work regarding human motion recognition has mostly focused on discerning different types of movement, such as walking, running.

In order to perform this type of classification, there are works which extract the basic movement of the subject and represent it as a simple 2D human stick figure and then, a predictive modular neural network time series classification algorithm is applied [4].

Regarding individuals with disabilities,the work that has been developed focus on analysing specific movements, in order to to allow the operation of instruments such as wheelchairs [5] [6].

B. Trajectory classification

In this work, points on the face of the subjects were detected over the frames. Then, based on their position we were able to extract their trajectory and use it to classify each one of them.

Trajectory classification has been an active research topics in several areas. However, there is no unique way to solve all problems as each technique is more suitable for each type of setup.

Most proposed methods for trajectory classification use hidden Markov models [7]. For instance, in the case of classifying human motion trajectories, where the trajectories were segmented at points of change in curvature and then hidden Markov models were used so as to classify them [8].

Other method commonly used relied on neural networks to classify this type of data. Such as the ones mentioned in the previous section.

Using Support Vector Machine (SVM) is another popular method used to classify trajectories. In some articles, the method proposed relied on extracting important features from trajectories, such as velocity, acceleration, orientation and classifying them with this algorithm [9] and [10].

Time-series classification can also be used to classify different trajectories. However, we are dealing with trajectories over the several frames and not exactly scalar values over the movie, so it is necessary to adjust the data and the models to fit each other [7]. One of the algorithms used to measure similarity between these features is the dynamic time warping. Then, based on the distance calculated between these time series it is possible to classify them based on their similarity [11].

1) Bag of Words: In this project, instead of using the methods previously stated, the Bag of Words (BoW) model is used.

The BoW model is a method commonly used in language processing and computer vision, in order to classify documents of text and images, respectively [12].

Regarding computer vision problems it is necessary to detect some sort of feature describing the images we are analysing. Therefore, the features selected should be in some way associated with the type of classification that is under study [13].

After defining these features, the similar features are grouped together using k-means clustering. Each cluster then corresponds to the ”codewords”. In the end, after grouping all these codewords, we have a ”codebook”, which is the analogous to a dictionary, where the different types of features are saved. With this information we can then use it to create an histogram later used to classify new images.

C. Points detection

In order to define the trajectories, it was necessary to identify key points of the face along the movie.

Therefore, one of the most important part of this project relied on the detection of the points on the face of the subjects, as this was the step which provided the data further used in the classification step.

In order to detect the face of the subject and the position of these facial features, one of several algorithms could have been used, among them the MTCNN, OpenFace or OpenPose.

In this thesis the OpenPose algorithm was used to detect these important features. This algorithm not only estimates facial landmarks, but is also able to estimate the position of joints of the human body [14] [15].

This detector is be able to provide us with 68 facial landmarks, plus another 3 provided by the body estimation, as can be seen in Fig.II-C.

However, not all the points provided by the detector were used, but only 19. One important aspect of the points selected is the fact that they could expressed the orientation of the face while preserving the rigidity of the shape.

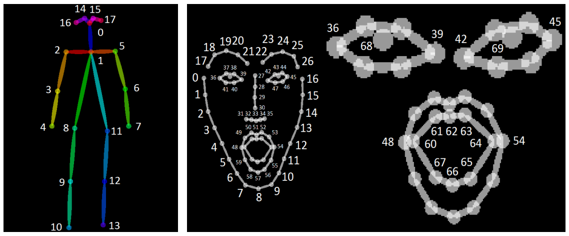


Fig. 1. Points captured by OpenPose.[16]

D. Point tracking

Another important topic in this project is feature tracking throughout the frames. This problem is referred as optical flow and, even though, it has been studied for many years it is still an open problem due to its complexity. Currently this is mostly solved by the Lucas-Kanade pyramidal algorithm.

This method consists on combining the luminosity information from several pixels nearby our point of interest, and trying to detect the new coordinates of the pattern previously identified [17].

This method would then allow to track the position of specific points throughout the sequential frames, as long as the points were visible. These should not move too fast and the contrast between the keypoints and their surrounding should be high.

III. Data Set

A. Subjects

The subjects analysed during this project were composed by different people both with and without disabilities. In order to acquire data from different subjects with some sort of disabilities, we went to Centro de Reabilitação de Paralisia Cerebral Calouste Gulbenkian, managed by Santa Casa da Misericórdia, where we are able to record 9 subjects whose maneuverability was affected differently. Another part of the recordings was done in Instituto Superior Técnico with some of their students, 13, who did not have any kind of disabilities.

All these subjects had to sign an informed consent form, in accordance to the World Health Organization.

B. Setup

In order to acquire our data, ideally all the subjects should be under the same conditions. This standardization would then allow to have better initial data, as we would not be so dependent on the configuration of the room, or where the cameras were placed relatively to the subject.

Overall, the environment used in order to record our movie consisted on the subject sitting on a chair, with a table in front of him. This table would have some objects properly spaced between them. These corresponded to our target, in other words, where the subjects would try to reach while performing their movements. Additionally, we also had a camera, Kinnect, in front of the subject so as to record their movements.

C. Types of Movement

Throughout this work it will be mentioned the existence of different types of movement. These movements allowed to compare and classify the subjects based on how these motions were performed. This way, since every subject was doing the same type of movements, we were able to segment the video for every subject. By doing this, as we analyse each movement, we were actually comparing the different ways to perform each specific movement.

Regarding the movements preformed, firstly the subjects were asked to look at the camera for a period of time, while trying not to move their head. Secondly, the subjects were asked to move to the object on their right side and return to the initial position, then repeat this but now moving to the left. Finally, the subjects were asked to move as close as possible to the object in front of them and then come back.

Between each movement the subject would always go back to the resting position, since this would allow to always have the same initialization, which is of extreme importance when comparing the different subjects. Additionally, this also helped to later segment the video without having overlapping movements.

IV. 3D Reconstruction

In this chapter the way the structure and orientation of the face of each subject was obtained will be explained.

In order to classify the different subjects, first it was necessary to detect the different facial landmarks throughout the movies, through one of the facial detectors previously explained in section II-C.

Then, from the points recovered we proceed to estimate the shape of each subject's head. Finally, the pose of the head, along the several frames, is estimated based on the position of the several points detected.

A. Data matrix

The data obtained and stored in the data set consisted on RGB videos with the subjects performing several predetermined motions. In order to obtain some data regarding those movies, we used the OpenPose algorithm. This information consisted on the coordinates of all the points detected throughout each video and it would be stored in a matrix that later would be analysed. This data matrix, W, is $2F \times P$ and is shown in equation (1).

$$W = \begin{bmatrix} u_{1,1} & \dots & u_{1,P} \\ v_{1,1} & \dots & v_{1,P} \\ \vdots & \ddots & \vdots \\ u_{F,1} & \dots & u_{F,P} \\ v_{F,1} & \dots & v_{F,P} \end{bmatrix} \tag{1}$$

In this case, the variable F corresponds to the number of frames and P the number of points tracked. More precisely we get the horizontal and vertical coordinates of each point for every frame, $\{(u_{fp}, v_{fp}) | f = 1, ..., F, p = 1, ..., P\}$ and then we combine and store it in W.

As this detection algorithm also provided the degree of confidence of each point, one could use this information and instead of storing all data available, only save the values with high confidence.

Due to the omission of points with low confidence, the existence of points not detected by OpenPose and occlusion possible occlusions, our W was only partially filled, as there was a high percentage of missing entries.

In Fig.2 it is shown an example of part of a data matrix which further will be used to show how the missing entries are filled.

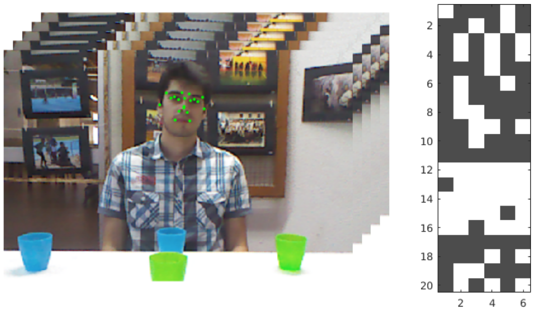


Fig. 2. Example of frame with points tracked (left), example of data matrix with correspondent missing (white) and known (grey) entries (right).

Therefore, we were left with a matrix with missing entries which we needed to fill in order to estimate not only the shape of the head of the subjects, but also the orientation throughout the frames.

B. Temporal constraint

As the data is acquired it is saved in the data matrix in a sequential way, so every 2 lines of the matrix we have the new coordinates of every point at the current frame. As the frequency of the frames was high it was possible to assume that in a small time period the velocity of the points was constant.

Therefore, in order to complete our data we considered that in the cases where some point was not detected only during 1 or 2 frames its coordinates could be the average of the known points. In other words we would simply assume that the projection of the 2D trajectory of the points was linear during this small interval, as it is represented in Fig.3.

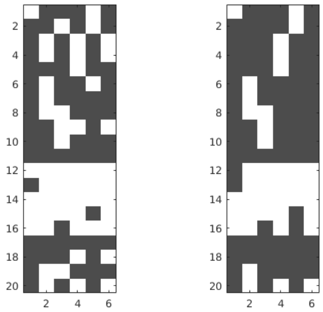


Fig. 3. Example of part of data matrix before (left) and after (right) the first completion step.

C. Shape matrix estimation

In order to further estimate the missing entries different methods could have been used.

In this problem, it was used a factorisation algorithm which is able to solve the structure from motion (SfM) problem while handling degenerate data and missing entries [18]. It was chosen this algorithm as it allows the estimation of the overall shape of the head of the subject, which would be used later.

This new algorithm is based on the Tomasi-Kanade algorithm, which is a factorisation method that allows us to recover the shape and motion of a given rigid object throughout a stream of images under orthographic projection [19].

In order to apply this algorithm, first it was necessary to centre our data matrix, W. In order to do that we subtract to each row its mean, $W_c = W - \frac{1}{P} W \mathbb{1}_{P \times P}$. In this case $\mathbb{1}$ corresponds to a matrix filled with 1 in all its entries. Therefore, the translation of the face throughout the frames is lost, and only the rotation is saved.

One of the key aspects of this matrix is the fact that it is rank deficient. Actually the data matrix can be factorised into two different matrices M and S, according to the expression $W_c = MS$.

In this case M is the Motion matrix, a matrix that encodes the camera rotation throughout the video. Additionally, M is $2F \times 3$ so, for every frame we have associated a 2×3 matrix, M_f, which encodes the camera rotation for the f^{th} frame.

The orthonormal vectors i^f and j^f point along the scanlines and the columns of the image of frame f, respectively, and are defined with respect to the world reference system. Therefore, with these vectors we can determine the orientation of the camera reference system at each frame.

Regarding S, this is a $3 \times P$ centred matrix which encodes the positions of the different points of object under study relatively to each other. This matrix has on each column the 3D coordinates of each one of the P points, which in this case is 19.

$$S = \begin{bmatrix} p_{1x} & \cdots & p_{19x} \\ p_{1y} & \cdots & p_{19y} \\ p_{1z} & \cdots & p_{19z} \end{bmatrix} \tag{2}$$

The estimation of S is of extreme importance, as this will help us to fill part of those missing entries.

In order to find these matrices we have to solve the minimisation problem presented in (3).

$$\begin{aligned} (M^*, S^*) = \underset{M,S}{\text{argmin}} \quad & \|W - MS\|_F^2 \\ \text{subject to} \quad & M_i M_i^T = \mathbb{I}_{2\times 2} \end{aligned} \tag{3}$$

Actually, this factorisation problem has closed form solution, which consists on applying the singular value decomposition (SVD) to the data matrix in order to decompose it and then obtain the estimations of both the M and S matrix.

As the images are expected to be orthographic projections, the effects of camera translation along the optical axis are not accounted for. Additionally, different distances to the camera of different points are also ignored as the model expects that the object is so far away that those distances are irrelevant, so everything is assumed to be on the same plane and which leads to degenerative data. However, the biggest problem relies on the amount of missing data.

One of the ways to cope with the existence of degenerative date relies on the addition of the scaling factor [20]. This new parameter α^f will correspond to the scaling of the shape for each frame, so different images at different distances can be acquired with no further problems.

Nevertheless the problem created by the existence of the missing entries, can only be solved through the use of algorithm 2 presented by [18]. In this case the overall idea is the same as the one presented above. However, this new problem does not have a closed form solution and needs to be solved iteratively.

To sum up, with this method one can easily factorise the data matrix and obtain both the shape matrix of our subject and also partially complete the data matrix. Partially complete, since throughout the project there were some frames with so little visible points that the coordinates of the missing points in these frames could not be determined by relying only on this tool. Therefore, only the frames with at least 6 points visible out of the 19 tracked were completed and used to estimate the overall shape matrix.

In Fig.4 it is shown the previous example of part of a data matrix before and after the completion step just described.

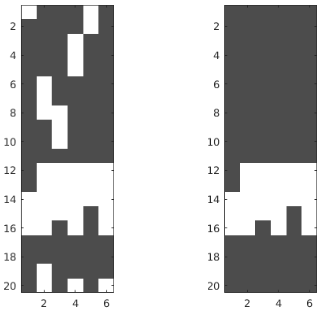


Fig. 4. Example of part of data matrix before (left) and after (right) the using the algorithm described

Below, in Fig.5, one can see the final result of the shape matrix obtained from one person, from different perspectives.

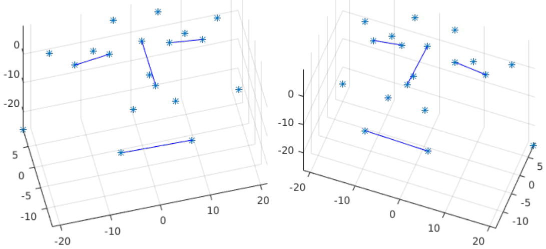


Fig. 5. 3D Shape matrix in different poses.

D. Motion matrix estimation

Even though part of W had already been filled and S acquired, it was imperative to acquire the position of the remaining points not yet detected, so as to estimate the motion along all the movie.

As it was possible to see, W was quite inhomogeneous. At the moment, this matrix either had big blocks completely filled with the expected entries, or blocks full of missing entries. In order to estimate the position of these points the Lucas-Kanade method for feature tracking, explained in II-D, was used for each block of consecutive missing entries.

The points previously detected in the completed frames were used as initialisation for the missing points we wanted to track. This means, that we would start in the known frames, and try to predict where the points had travelled based on the optical flow equations.

After tracking the new positions of the missing data then we would make use of S, obtained in the previous section and impose it into the new points. In other words, we wanted to find the best approximation of the data points, which actually also shared the same structure as the Shape matrix. This was quite important since it allowed to preserve the shape of the user's head.

Actually, this imposition problem consists on solving an orthogonal Procrustes problem with scaling [21]. This can be expressed as finding the matrix, Ω, that was able to align both S and our data, as shown in minimisation problem presented in the equation (V-A1).

$$\begin{aligned}(\beta^*, \Omega^*) = \underset{\beta,\Omega}{\text{argmin}} \quad & \|\beta\Omega S - X\|_F^2 \\ \text{subject to} \quad & \Omega\Omega^T = \mathbb{I}_{2\times 2} \\ & \beta > 0\end{aligned} \tag{4}$$

Where X corresponds to the centred coordinates of the points in study and has dimensions of $2 \times P$.

This problem has already been solved and its closed form solution actually comes down to computing a SVD and using both singular vectors to calculate Ω, $\Omega = U_{2\times 2}V_{3\times 2}^T$, and the average of the singular values would correspond to β.

Therefore, by knowing Ω and β we can estimate the best position for the points given a certain shape. This way we were able to get a more realistic approximation of the points we were trying to track which we can then use to help us fill our data matrix.

One of the biggest disadvantages of this procedure relies on the fact that as the number of frames analysed increases we are subjected to higher error. In order to go around this problem the same idea was used twice. More precisely, the points were tracked along the normal appearance of the frames, and then tracked again, but backwards, so starting in the last frame, and finishing on the first one. The coordinates of the tracked points would then correspond to a weighted average of these two results.

As well as in section IV-B, this was only possible to apply, due to the fact that W was organised with the frames in accordance with the natural flow of time.

After running this procedure in each block of missing entries we finally reached our goal of having our data matrix W completed. Then it was possible to estimate W over all the frames of the movie.

V. Movement Classification

In this section, it is explained the way how we extract features from the subjects and later how these are used not only to classify the different subjects, but also to create a classifier.

The BoW model was used in order to classify the subjects. However, in order to use this method it was necessary to describe our data in a simpler way. Therefore, two different types of features were developed in this thesis, which aim to encode the trajectory of the users head. These features were independent of the subject, so our data had to be pre-processed before being encoded into those descriptors.

A. Pre-Processing Data Matrix

In this section the pre-processing of our data is further discussed.

In order to compute the trajectons further described we needed to make the data matrices invariant to the shape matrix of each subject. This was achieved by defining a reference shape and then finding the transformation between this one and the shape associated with the data matrix under analysis.

As the transformation between this shapes is defined, we can adapt the data matrix in order to remove its influence.

1) Reference Shape: In order to remove the influence of the subject's head, a Shape matrix of reference was created, S_{ref}. Then, every S was mapped onto the one of reference, and later, this linear transformation would be used to adapt M, and normalise W.

Regarding S_{ref}, it was used a S from one subject, where tracking went particularly well. Then this matrix was normalised so as to have its first singular value equal to 1.

After defining our standard shape matrix, we had to find the linear transformation between these matrices, which would encode both the scaling and rotation between them. This problem was similar to the one presented in the minimization problem . But now, instead of a scaling factor a 3×3 diagonal matrix, D, was defined, since the shape of the head can vary in different directions.

Therefore, in order to determine these new matrices we would then need to solve the optimization problem presented in equation (5).

$$\begin{aligned}(R^*, D^*) = \underset{R,D}{\text{argmin}} \quad & \|RDS_{ref} - S\|_F^2 \\ \text{subject to} \quad & RR^T = \mathbb{I}_{3\times 3} \\ & D_{ii} > 0 \\ & D_{ij} = 0,\ i \neq j \\ & i = 1,2,3 \text{ and } j = 1,2,3\end{aligned} \tag{5}$$

Actually, this problem consists on an anisotropic Procrustes with post-scaling. However, according to [21], this new problem does not have a closed form solution, but can still be solved iteratively

By repeatedly updating both D and R in this algorithm, it was possible to determine their final values and with them we finally had the required elements to transform S_{ref} into S.

2) Standardise Data: As the shape has been standardised, then we needed to update each subject's W, so as to remove the influence of the subject's S.

Regarding D, this matrix was used to get the S_{ref} with the same dimensions as S. However, we want to focus on the orientation of the head, so this term was neglected.

The relative distance between the subject and the camera itself was reflected in the scale factor, previously described in section IV-C. In the first frame of the different recordings, the subjects were approximately at the same distance to the camera. Therefore, the scale factors along the video were normalised based on this value. This way, while the subjects were on the resting position the scale factor would be around 1 and as they move closer to the camera, this value would increase accordingly. These values were saved in a column matrix, Λ.

Regarding the orientation, by multiplying both M with R, we got a matrix, M_{norm}, whose properties were similar to the ones of M, while having its rotation adapted to S_{ref}.

Therefore, we were able to reformulate the data matrix of each subject, as $W_{norm} = M_{norm}S_{ref}$. This new data matrix was normalised, only containing the orientation of S_{ref} and the relative scale factor was stored independently on Λ.

B. Features

After normalizing the data matrix we were left with the relative position of the points of interest over the several frames, as shown in Fig.6.

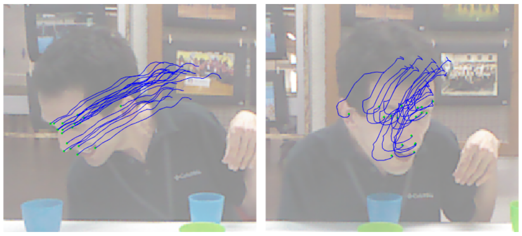


Fig. 6. Examples of position of the points over several frames, while the subject with disabilities was going to the left and forward

In order to classify this data set, two different features, further referred as trajectons, were developed and tested. These mainly focused on analysing the orientation of the head's shape in two different ways. Whereas the translation along both perpendicular axis between the camera and the subject was not taken into consideration.

For each subject several blocks of frames would be created, according to the sliding window method, until all the movie had been covered. In the end, each one of these blocks would lead to an independent trajecton.

1) 6P feature: This feature consisted on creating a discretized 3D cube, where we would project the reference shape according to the pose of the subject each frame. This cube would be voxelized and each voxel would be scored as the facial landmarks passed by them. Overall this feature would focus on the spatial distribution of the points over the frames.

Therefore, this feature focus more on the space occupied by the points over the frames. Regarding this special cube that was designed, in the first two axis, the position of the points tracked was stored and the final axis would store the scale factor.

Additionally, not all the 19 points tracked were used in this configuration. Since we did not label the score of the voxels then we had no way to know which point passed at each place, so we only use 6 points.

As the voxels are inside the cube it is reasonable to assume that neighbouring voxels have similarly meaning. However, as we pass this information to a vector their relationship is lost. Therefore, in order to obtain a more robust descriptor, the score of the adjacent points were also improved. By increasing the score of the neighbouring voxels, we induce a connection between them, that later on during clustering can help to better classify the different subjects.

In Fig.7, one can see the descriptor presented of a single frame.

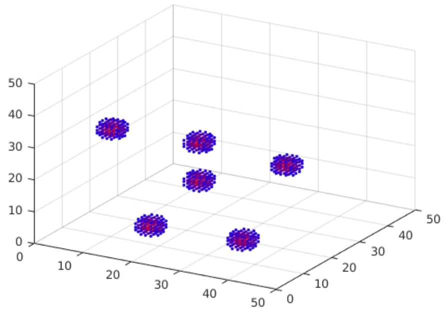


Fig. 7. Representation of the 6P trajecton of one frame.

2) Shaky feature: This second feature focused on analysing the degree of the oscillations in the different movements. On the one hand, one would expect that in the case of subjects without disabilities their movement would be fluid with little to no shacking, so the rate of change would be essentially constant.

On the other hand, people with disabilities have lower control over their movement. Therefore, while they try to perform a certain action, there are still a lot of involuntary movements. So, the movement performed by these subjects will have the tendency associated with the movement requested, as well as several other random changes in other directions.

Therefore, this feature focus more on the temporal changes of the coordinates of the points.

In this descriptor each frame would represent each point as a pair value, where both values would be between -4 and 4. This would correspond to the change of coordinates between two frames of each point. More precisely, a positive value would be associated with an increment between the coordinate of the specific point from a specific frame in a certain direction, 0 in case the point did not move substantially and a negative value when the coordinate of the point would decrease. Therefore, using all the combinations of possible directions, each point on each frame could follow one of 81 trajectories.

This increment meant that instead of only detecting the direction of the movement of the point based on the signal, we were also able to know by how much it actually moved.

In order to encode the rate of change of the scale factor, a similar process was applied. As this value is referent to the shape as a whole, we only had nine possible cases which would encode whether this value increased or decreased and the intensity of this change.

Each frame would store the changes of its points and by combining several frames we would be able to get our feature, as shown in Fig.8.

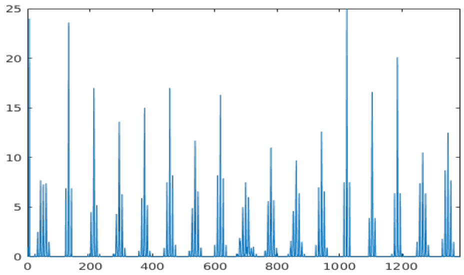


Fig. 8. Representation of the Shaky trajecton.

C. Classification

After creating the different features associated to each subject, we were able to create our codewords. These would correspond to the centres of the clusters formed through k-means clustering. This clustering algorithm was applied independently to each type of Segment analysed, so that different movements would have different clusters associated.

In order to classify a new subject, their trajectons were compared with the clusters stored on the respective codebook and each trajecton was then associated with the closest cluster. With this information, we constructed an histogram of the subject under analysis, reflecting the percentage of trajectons associated with each cluster.

As we already knew the subjects classification of the training set, for each segment, we could find the closest one using the nearest neighbour algorithm and assign the same label.

In the end, this process was repeated twice, once for each type of feature presented in V-B.

VI. Results

A. Filling missing data

As it was explained in section IV the data matrix obtained from the facial landmark detector was filled with missing entries. Subjects without disabilities would have about 40% of missing entries. In contrast, the other type of subjects would have a value around 55%. In Fig.9 one can see the distribution of missing entries of the raw data of two different subjects.

By applying the procedure described in section IV, after completing the predictable missing entries and using the algorithm proposed by [18] we could further complete our target matrix and build the Shape matrix, as shown in Fig. 9.

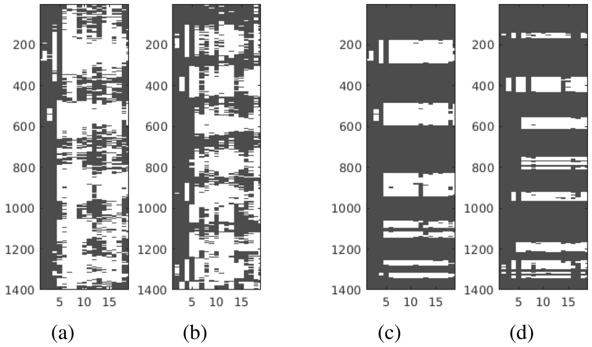


Fig. 9. Original data matrices of a motor-impaired person (a) and a person without any disability (b). Regarding (c), corresponds to the same matrix as (a) after completing its entries with the factorization algorithm and (d) is the same, but associated with (b). In this case the white area corresponds to missing entries and the grey one to the entries with values obtained from OpenPose.

After this completion step, only frames with too little points detected would still be incomplete. Moreover, these formed different types of gaps throughout the data matrix. Some of them would consist only of 3 frames, but others would be longer than 50 frames.

Regarding the smaller gaps, the method proposed was quite reliable, it would easily detect the new position of the points of interest and properly adjust them to the shape matrix. However, the larger ones were quite alarming, since we were bound to accumulate errors. Nevertheless, due to the fact that we run the Lucas-Kanade algorithm in the two directions, and also with the help of the imposition of S, we were able to successfully complete all the gaps independently of their size.

B. Trajectons

The images presented in Fig.10 correspond to an example of the points tracked during the resting position of two different types of subjects. However, one can already see clear differences between the two, as an increasingly control of the movement led to smaller trajectories.

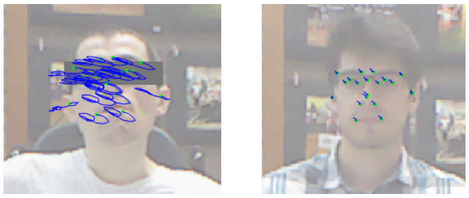


Fig. 10. Different subjects during the resting position along with the estimation of their face's points during the last 25 frames. In this example a subject with high (left) and no (right) physical limitations are shown.

Another important segment analysed was the forward movement shown in Fig.11.

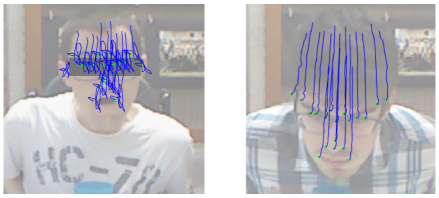


Fig. 11. Different subjects while reaching the object in front of them, along with the estimation of their face's points during the last 25 frames. In this example a subject with high (left) and no (right) physical limitations are shown.

1) 6P Trajectons: As previously explained in subsection V-B1, this type of feature creates a discretized cube which gets each entry scored as the points pass on each voxel.

Regarding the resting position, as it is possible to see in Fig.12, one can say that it is harder for the subject with disabilities to completely restrain their movement. Firstly, the cloud originated by the position of the point tracked is broader when compared to the second subject, as well as less intense. So, we can confirm that the this subject did not spend too much time in the same position. However, in the case of the second subject not only the size of the cloud is smaller, but we can also see higher intensity in the centre, which means that stability was more easily achieved by this subjected.

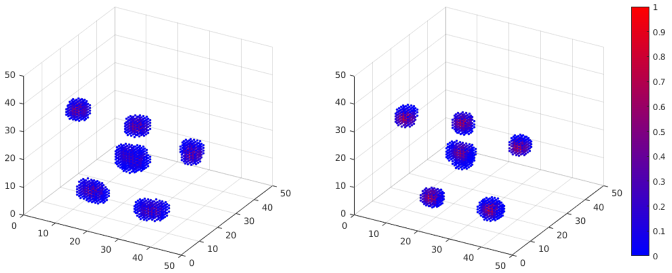


Fig. 12. Full 6P trajecton, over 25 frames, during the Stable movement, associated with a person with disabilities (left) and without (right).

In the case of moving forward, we can detect different tendencies. In the case of the subjects with high physical limitations, one can see a movement with low amplitude and highly irregular. In contrast, the movement of subjects without any physical limitation is clear and the changes occur mainly along the scaling factor.

Overall this type of trajecton seemed quite good and the simple fact of being able to discern the different types of subjects just by looking at the feature is quite promising.

2) Shaky Trajecton: In this second feature, previously explained in subsection V-B2, we focus on the changes of the coordinates of each point as the head of the subject rotates. This trajecton consists on a plot with several spikes, and their size is intrinsically associated with the amount of time the rate of change of that specific point was constant.

Regarding this trajecton, as we analyse the resting position it is possible to predict in some way the resulting trajectons for the different subjects. In the case of the subjects with no disabilities, as they can better control their movements then they will stay immobile more easily. Therefore, this type of subjects would have bigger spikes since the output would be always constant. As for subjects with some physical limitations, these were not able to restrict as easily their movements. So, as they try to stay immobile, the unavoidable shacking of the head leads to a trajecton with more random entries different. Below, in Fig.14, it is possible to see the pattern described, just as predicted.

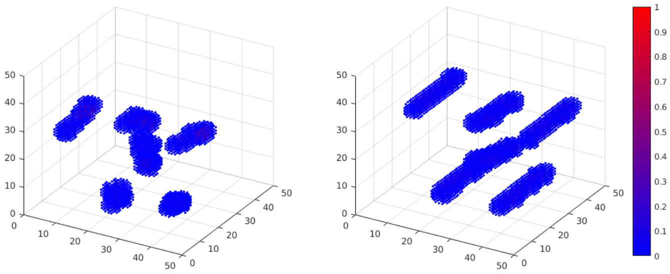


Fig. 13. Full 6P trajecton, over 25 frames, while moving forward, performed by a person with high (left) and no (right) physical limitations.

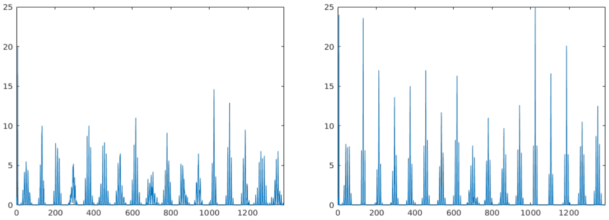


Fig. 14. Example of a Shaky trajecton as a subject with (left) and without (right) disabilities try not to move.

Intuitively, one would expect that independently of the type of movement, subjects without any type of physical disabilities would have their movements more fluid. This fact would imply that their points trajectories would be constant along consecutive frames. In Fig.15, it is shown the resulting trajecton for both subjects as they move forward. According to the previous statement, we can see the size of the spikes in accordance with the type of subject.

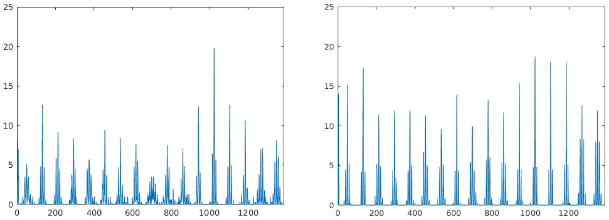


Fig. 15. Example of a Shaky trajecton as a subject with (left) and without (right) disabilities try to move forward.

As one can see by the information presented here, this trajecton is quite promising, as it emphasises one of the differences in the behaviour of the subjects while they perform the given tasks.

C. Clustering

1) 6P Trajecton: After analysing different number of clusters, it was concluded that there was no optimal value for this variable.

In Fig.16, it is shown the results after clustering our data, for two types of movement, with different values of k, $\{4, 6\}$. In each row one type of movement is analysed (first the stable movement, and then the moving forward) and in each column a different number of clusters is used. On each parcel it is shown an image representing the distribution of the subject's trajectons, while they were doing a certain movement. Actually, each line corresponds to a subject histogram and each column with a cluster. Brighter colour corresponds to a higher percentage of that subject's trajectons associated with a certain cluster. The first subjects (until first red line) don't have any disability, second group of subjects have some physical disabilities and in the last one are the subjects with severe physical disabilities

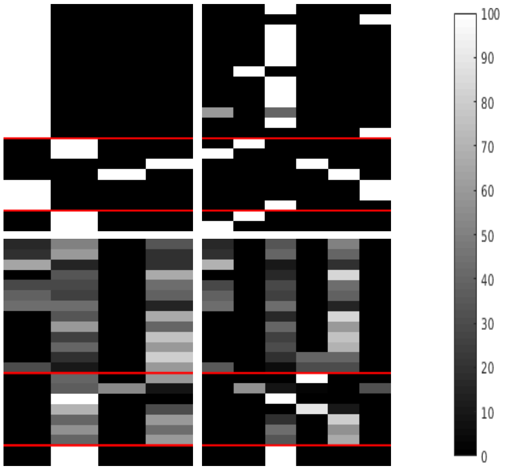


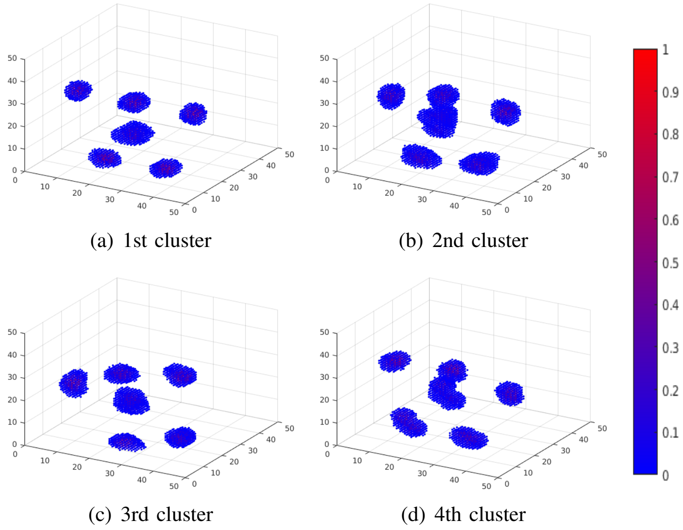
Fig. 16. Clusters obtained from 6P trajecton

In the case of the Stable movement with 4 clusters, there is one single cluster which mainly expresses all the individuals without disabilities, while the remaining clusters better represent the subjects with disabilities. Therefore, we can conclude that there are some clusters highly associated with the subjects without disabilities, whereas the remaining clusters are usually associated with the remaining subjects.

Regarding the subjects with disabilities analysed they were quite different between them. No one had the same problem and the same physical limitation. Therefore, while we were grouping them as a certain type of subjects, depending on the task asked to perform, they would show different behaviours, in some cases even similar to subjects without any physical limitation.

Therefore, more than detecting whether the subject had physical disabilities or not, we are analysing how the subjects can perform under different situations. In the feeding robot scenario this type of information would be extremely helpful, as it would be possible to adapt different states of the feeding process to the subjects themselves.

Below in Fig.17, the clusters obtained for the Stable movement with 4 clusters are shown.



(a) 1st cluster

(b) 2nd cluster

(c) 3rd cluster

(d) 4th cluster

Fig. 17. Clusters of the 6P trajectons obtained from k-means algorithm with the Stable data while using 4 seeds.

The clusters shown on this figure correspond to the ones by clustering the Stable data set with 4 seeds. The representation of the clusters describes quite well our data. Regarding the first cluster, if we analyse the result from Table 16, we can conclude that this one is associated with subjects without any physical limitations. The second cluster, as one can see, encloses the subjects who were not able to stabilize their head as well. In the third cluster, we can see that this one is associated with subjects who were able to maintain the head reasonably stable, while having their head pending to one of the sides. On the other hand, in the fourth cluster we see a similar pattern, but in this case the subjects have their head pending to the opposite side.

2) Shaky Trajecton: Similarly to the 6P trajecton, the same procedure was applied to the Shaky trajecton. In Fig.18, we have a similar table, with the same movements and number of clusters shown, but now for the Shaky trajecton.

Overall the results obtained are the same as in the 6P trajecton. We can detect some clusters predominantly associated with subjects without disabilities and others to with the ones with the motor impaired ones.

However, as one can in Fig.19, in this case, the clusters formed with this trajecton are not so intuitive.

As it is possible to see, the output provided in the first cluster, associated with subjects without disabilities, has almost all points with high peaks as we would expect. The opposite is found on the fourth cluster where the subjects had a clear difficulty controlling their movements.

D. Classification

As the clusters have been defined, it is now possible to classify new subjects. In order to simulate this step, we simply divided our data set into a training and a test set.

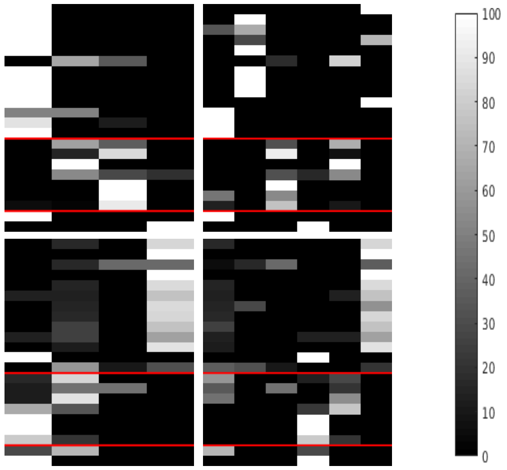
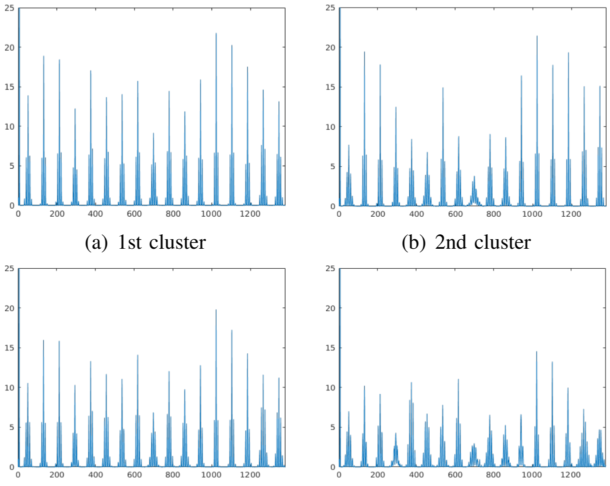


Fig. 18. Clusters obtained from Shaky trajecton



(a) 1st cluster

(b) 2nd cluster

(c) 3rd cluster

(d) 4th cluster

Fig. 19. Clusters of the Shaky trajectons obtained from k-means algorithm with the Stable data while using 4 seeds.

In order to create these two sets 2 subjects without disabilities and 2 subjects with disabilities were chosen at random and formed the test set. The remaining ones were associated with the training set. After testing 20 combinations of test sets, on average these subjects would be classified accordingly 65.3% of the times with the 6P trajecton and 66.5% with the Shaky trajecton.

Most of the misclassification fall on classifying subjects as one without disabilities, when in fact that was not the case. One of the reasons can be due to the fact that as there are so many different types of subjects with disabilities and their number is so low, then there are not enough subjects similar to one another to allow a proper classification.

Another reason relies on the fact that each subject had different disabilities. So, in some cases the subject removed might not have another one with a similar behaviour on the test set, so there was no cluster close to his data. This last problem could be solved by increasing our data set. More specifically, by getting more data from different subjects with similar disabilities.

A new test was done, but now instead of completely removing the subjects of the test set from the training set, only a random part of their movement was removed. Different combinations subjects and parts removed were tested and in the end, on average, the subjects without disabilities would be classified accordingly 71.5% of the times with the 6P trajecton and 93.1% with the Shaky trajecton.

VII. Conclusions

In this thesis we focused on creating a methodology, which aimed to classify different subjects based on their disabilities.

As there was no type of data set available one had to be constructed for this project. This data set consisted on labelled recordings of different subjects.

In order to classify our subjects, two different types of features were developed. One of them would focused more on the spatial information obtained through the recordings, and the other on the temporal information.

These features were estimated based on the 2D projection of the trajecton of facial landmarks. As the position of these were detected on each frame, they were stored onto a data matrix. However, this data matrix was filled with missing entries, as the current facial landmark detectors would fail when facing occlusions or were simply not accurate enough. Therefore, a large part of this work consisted on estimating the values of these missing entries.

After acquiring this data matrix and computing our trajectons, our new data was clustered using k-means clustering. Afterwards, we were able to classify our subjects based on the nearest neighbour algorithm, which was able to provide reasonable results. However, due to the low data quantity and the variability of the disabilities of the subjects analysed, better results were not achieved.

Overall one of the greatest draw backs in this project relied on the lack of quantitative baselines. To the extent of our knowledge no work facing this problem has been done before, so there were no benchmarks, nor ground truth, nor data sets available to properly compare our work with. Therefore, most of the decisions had to be done based on a qualitative analysis which might lead to some biased results.

Nevertheless, the features presented in this thesis were still quite promising.

References

[1] WHO, "World Report on Disability - Summary," *World Report on Disability 2011*, no. WHO/NMH/VIP/11.01, pp. 1–23, 2011.

[2] S. Gulati and V. Sondhi, "Cerebral Palsy: An Overview," *The Indian Journal of Pediatrics*, nov 2017.

[3] C. Morris and D. Bartlett, "Gross motor function classification system: impact and utility," *Developmental Medicine & Child Neurology*, vol. 46, no. 1, pp. 60–65.

[4] V. Petridis, B. Deb, and V. Syrris, "Detection and identification of human actions using predictive modular neural networks," in *2009 17th Mediterranean Conference on Control and Automation*, June 2009, pp. 406–411.

[5] M. Shweta, K. Yewale, M. Pankaj, and K. Bharne, "ARTIFICIAL NEURAL NETWORK APPROACH FOR HAND GESTURE RECOGNITION," 2018.

[6] S. Yokota, H. Hashimoto, Y. Ohyama, J. She, D. Chugo, and H. Kobayashi, “Classification of body motion for human body motion interface,” in *3rd International Conference on Human System Interaction*, May 2010, pp. 734–738.

[7] J. Lee, J. Han, X. Li, and H. Gonzalez, “Traclass: Trajectory classification using hierarchical region based and trajectory based clustering,” *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 1081–1094, 1 2008.

[8] F. I. Bashir, A. A. Khokhar, and D. Schonfeld, “Object trajectory-based activity classification and recognition using hidden markov models,” *IEEE Transactions on Image Processing*, vol. 16, no. 7, pp. 1912–1919, July 2007.

[9] X. Xiao, H. Hu, and W. Wang, “Trajectories-based motion neighborhood feature for human action recognition,” in *2017 IEEE International Conference on Image Processing (ICIP)*, Sept 2017, pp. 4147–4151.

[10] A. Boubezoul, A. Koita, and D. Daucher, “Vehicle trajectories classification using support vectors machines for failure trajectory prediction,” in *2009 International Conference on Advances in Computational Tools for Engineering Applications*, July 2009, pp. 486–491.

[11] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana, “Fast time series classification using numerosity reduction,” in *In ICML06*, 2006, pp. 1033–1040.

[12] M. Mctear, Z. Callejas, and D. Griol, *The Conversational Interface*, 2016.

[13] L. Fei-Fei and P. Perona, “A bayesian hierarchical model for learning natural scene categories,” in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05)*, vol. 2, June 2005, pp. 524–531 vol. 2.

[14] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, “Realtime multi-person 2d pose estimation using part affinity fields,” 2017.

[15] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh, “Convolutional pose machines,” 2016.

[16] “OpenPose — OpenPose Output.” [Online]. Available: https://github.com/CMU-Perceptual-Computing-Lab/OpenPose/blob/master/doc/output.md

[17] S. Baker and I. Matthews, “Lucas-Kanade 20 years on: A unifying framework,” *International Journal of Computer Vision*, vol. 56, no. 3, pp. 221–255, 2004.

[18] M. Marques and J. Costeira, “Estimating 3D shape from degenerate sequences with missing data,” *Computer Vision and Image Understanding*, vol. 113, no. 2, pp. 261–272, 2009.

[19] C. Tomasi and T. Kanade, “Shape and motion from image streams: a factorization method.” *Proceedings of the National Academy of Sciences*, vol. 90, no. 21, pp. 9795–9802, 1993.

[20] C. J. Poelman and T. Kanade, “A paraperspective factorization method for shape and motion recovery,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 3, pp. 206–218, March 1997.

[21] J. C. Gower and G. B. Dijksterhuis, *Procrustes Problems*, 2004.