UNIVERSITAT POLITÈCNICA DE CATALUNYA

MASTER'S THESIS

MASTER'S DEGREE IN AUTOMATIC CONTROL AND ROBOTICS

Non-Rigid Structure from Motion for Complex Motion

Author:

Sergi Molina

Supervisor:

Dr. Antonio Agudo

Dr. Francesc Moreno-Noguer

October, 2016

ESCOLA TÈCNICA SUPERIOR D'ENGINYERIA INDUSTRIAL DE BARCELONA

Contents

Abstract

Recovering deformable 3D motion from temporal 2D point tracks in a monocular video is an open problem with many everyday applications throughout science and industry, or the new augmented reality. Recently, several techniques have been proposed to deal the problem called Non-Rigid Structure from Motion (NRSfM), however, they can exhibit poor reconstruction performance on complex motion. In this project, we will analyze these situations for primitive human actions such as walk, run, sit, jump, etc. on different scenarios, reviewing first the current techniques to finally present our novel method. This approach is able to model complex motion into a union of subspaces, rather than the summation occurring in standard low-rank shape methods, allowing better reconstruction accuracy. Experiments in a wide range of sequences and types of motion illustrate the benefits of this new approach.

Keywords: Non-Rigid Reconstruction, Clustering, Low-Rank Models, Monocular Vision.

Chapter 1

Introduction

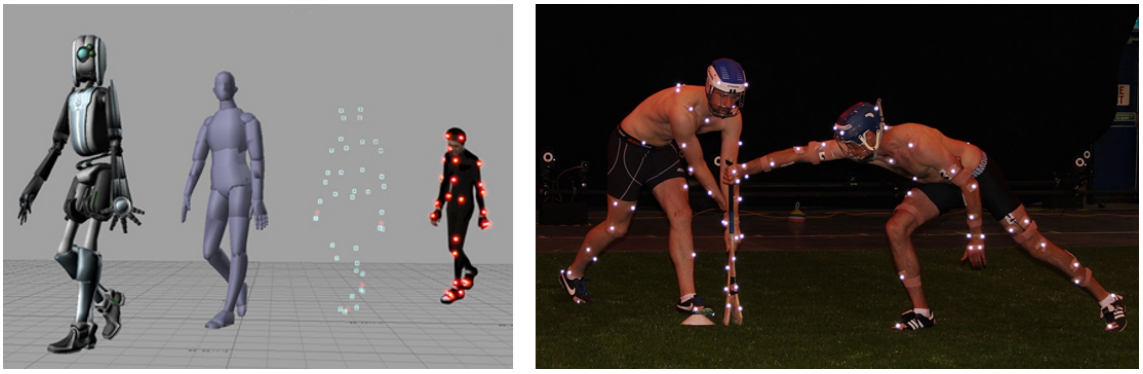

Figure 1.1: Two scenarios where the non-rigid reconstruction could be useful. On the left there is a person being tracked with several points marked in red, which lately those points have been transfered to a virtual space overlapping a robot shape which follows the same actions the actor performed. On the right two field hockey players are being tracked to monitor their activity and evaluate the performance [1]. Currently, these models are acquired with complex and expensive motion capture systems, based on multi-camera systems. In contrast, we propose a method that only uses a monocular camera.

Reconstructing the 3D shape of objects from a monocular image sequence has been an extremely important research area in computer vision for decades, with many everyday applications in robotics, augmented reality and medical imaging. The rigidity prior has proven to be a powerful constraint to solve the problem with Structure-from-Motion (SfM) algorithms, producing practical and robust solutions [2]. However, rigid SfM methods fail when applied directly to deformable objects, such as the human body or smiling faces.

To solve this problem, Non-Rigid Structure from Motion (NRSfM) algorithms have been proposed. In this case, the joint estimation of non-rigid 3D shape and camera pose parameters from 2D point trajectories in a monocular video, normally results in a non-convex optimization problem, and the orthogonality constraints on the pose parameters makes the problem more complicated. During acquisition, the monocular camera observes a deformable

object while performs a rigid motion (a combination of translation and rotation) around the body we want to reconstruct. Since many different 3D shapes can have similar image observations, the reprojection constraints are not sufficient to obtain a single solution, and most works use additional a priori knowledge about the camera motion and the deformation of the object.

This is an issue not only studied in the computer vision field, but also in other disciplines like computer graphics, sports or video-games, where the actors could be tracked (even using natural landmarks) and then overlap a virtual avatar using the same motion (see Fig. 1.1-left). Human-computer interaction or biometrics are other fields where the model recovered from a human body could be used to evaluate the performance during a sport match, whether the technique used in an action is correct or not, or if a person is suffering a medical issue, comparing the motion with other patterns (see Fig. 1.1-right).

In this project a brief review of the state of art is done, focusing on the Dai *et al.* [3] approach, seeing that only applying this method is not enough in terms of accuracy when dealing with complex motions. So in addition to Dai *et al.* [3] method is used the low-rank representation by Liu *et al.* [4] to cluster the 3D non-rigid motion into a union of subspaces, and the recent work proposed in [5]. Combining both of them is possible to recover 3D shape of non-rigid bodies from monocular video with a low degree of error as it can be seen in the experiment results.

In chapter 2 is covered the problem we are facing, defining the main formulation we are working with during the project, then some theory as the state of the art, for solving the NRSfM in single subspaces and the recovery of subspace structure, are presented in chapters 3 and 4, respectively. In section 5 are showed the experiments performed to check the accuracy of the algorithm regarding the clustering and 3D shape reconstruction. The results obtained are compared with state-of-the-art techniques to measure the improvement ratio if any, of our approach. Finally, we have elaborated a temporal planning covering all the stages, and also calculated the costs associated to the project development. The conclusion together with the future work, can be found in the last section 7.

1.1 Related Work

NRSfM is an inherently ill-posed problem unless additional a priori knowledge of the shape and camera motion is considered. A seminal work by [6] proposed a low-rank shape con-

straint as an extension of the [7] factorization algorithm to the non-rigid case. Their key insight was to model time-varying shape as a linear combination of an unknown and rigid shape basis under orthography. Although this prior has proved to be a powerful constraint, it is insufficient to solve the inherent ambiguities in NRSfM. It was shown in [8] that the low-rank shape prior in addition to orthonormality constraints on camera motion are sufficient for noise-free observations. Recently, [3] imposed the low-rank shape constraint directly on the time-varying shape matrix via a trace norm minimization approach. These priors can be computed using either principal component analysis over training data [9, 10], applying modal [11, 12] or spectral [13] analysis over a rest configuration, or they are estimated on-the-fly [6, 14, 15, 16].

Most approaches have required the use of additional priors using different optimization schemes to include temporal smoothness [17, 18, 19, 14, 16], smooth-time trajectories [20, 21], inextensibility constraints [10], rigid priors [22] and spatial smoothness [23, 16]. Bundle adjustment has become a popular optimization tool for refining an initial rigid solution [7] optimizing the camera pose, shape basis and coefficients while incorporating both motion and deformation priors [19, 14]. Other approaches have modeled deformation using a low-rank trajectory basis per 3D point [20], enforcing smoothness on their paths [21] or considering a force basis with physical interpretation [24].

In this work, we propose to explore the concept of complex motion, to process scenarios that can not be well-modeled by a single linear subspace. To achieve this, we present an algorithm that recovers simultaneously the 3D reconstruction of a complex scenario while obtains its temporal clustering, allowing us to know the different primitives the object is performing.

Chapter 2

Problem statement

The NRSfM problem aims to recover an object's 3D shape from the corresponding 2D points obtained with a monocular camera, tracking relevant features from whichever deformable body, like a face, or a human body, moving along a sequence of frames involving a complex motion[1]. For complex, we mean a movement which can be described as a summation of simpler or primitive actions. For a face the primitives could be for example shout, cry and smile, and for a human body could be jump, run, sit or dance (see Fig. 2.1). These complex movement increase the difficulty of the original problem, which only considers a single subspace/primitive action, so in order to improve the reconstruction accuracy we must detect and cluster the simple actions involved along the sequence for a proper handling.

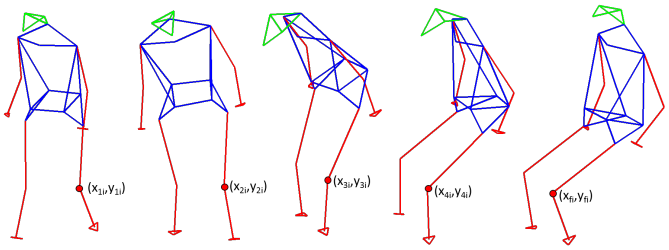

Figure 2.1: Example of a small portion from the sequence "$p4\_table$" from the UPMC dataset, showing a human body performing different motions. In this case, the person goes from walking, to stand and finally sitting in a table with the calves hanging. We show the evolution of a generic point i.

[1]The process of obtaining these tracked points is not going to be explained in this paper, being out of the paper scope, so during the experiments we work with a series of prerecorded data coming from different datasets.

2.1 Formulation

As we have said, all starts with a matrix containing all 2D points from a body recorded from a monocular camera. This matrix is build stacking all the coordinates points $[x, y]$ of all the points being tracked for all frames, obtaining in this manner the $\mathbf{W} \in \mathbb{R}^{2F \times N}$ matrix, where N represents the number of points and F the number of frames. In Fig. 2.1, we have an example of the person's knee 2D coordinates being tracked along the frames. Number of tracked points can be as high as we want (from sparse to dense cases), but we have to take into account that as we increase the number, the computational cost of computing the algorithm also rises. In this project, we work with sequences in range between 35 and 50 points approximately, since our aim is not only to compute the 3D reconstruction but also the temporal clustering into primitives. Coming back to Fig. 2.1, apart from the knee, there also points for the feet, hips, torso, head and arms, giving a total of 37 points to form a human body structure. The observation matrix can be ordered as:

$$\mathbf{W} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ y_{11} & y_{12} & \cdots & y_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ y_{21} & y_{22} & \cdots & y_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{f1} & x_{f2} & \cdots & x_{fn} \\ y_{f1} & y_{f2} & \cdots & y_{fn} \end{bmatrix}.$$

Assuming an orthographic camera model (i.e., all the projection lines are orthogonal to the projection plane), we can write the projection of the 3D points $\mathbf{X}_f$ onto the image plane as a $2 \times N$ matrix:

$$\mathbf{W}_f = \mathbf{R}_f \mathbf{X}_f, \tag{2.1}$$

where $\mathbf{R}_f$ are the first two rows of a full rotation matrix. Note that in previous projection equation, the observations have been already centralized, i.e., we have deleted the translation part from the equation, the camera motion matrix it is reduced to pure rotation.

If we now consider the F image frames, we have to define the $\mathbf{R}$ matrix that takes the form $\mathbf{R} = blkdiag([\mathbf{R}_1\, \mathbf{R}_2\, \cdots\, \mathbf{R}_f]) \in \mathbb{R}^{2F \times 3F}$ where each $\mathbf{R}_f$ is the rotation camera matrix for each frame f. The Eq. (2.1) then becomes for all image frames:

$$\mathbf{W} = \mathbf{R}\mathbf{X}. \tag{2.2}$$

The time-varying shape matrix $\mathbf{X} \in \mathbb{R}^{3F\times N}$ (see Eq. (2.3)) can be modeled as a linear combination of K shape bases $\mathbf{B}_k \in \mathbb{R}^{3\times N}$ with shape coefficients c_k. So the expression $\mathbf{X}$ can be described as $\mathbf{X} = (\mathbf{C} \otimes \mathbf{I}_3)\mathbf{B}$, where $\mathbf{B} \in \mathbb{R}^{3K\times N}$ stacks the K shape basis and $\mathbf{C} \in \mathbb{R}^{F\times K}$ includes the corresponding weight coefficients. $\otimes$ denotes the Kronecker product and $\mathbf{I}_3$ is the 3×3 identity matrix. Many works [19, 14, 16] use this factorization to compute $\mathbf{X}$ indirectly, i.e., they set some predefined special shape bases, so then they can find $\mathbf{X}$ with the relation.

$$\mathbf{X} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1f} \\ Y_{11} & Y_{12} & \cdots & Y_{1f} \\ Z_{11} & Z_{22} & \cdots & Z_{1f} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nf} \\ Y_{n1} & Y_{n2} & \cdots & Y_{nf} \\ Z_{n1} & Z_{n2} & \cdots & Z_{nf} \end{bmatrix} \in \mathbb{R}^{3F\times N}. \tag{2.3}$$

2.2 NRSfM by Nuclear Norm Minimization

Inspired by [3], we do not make use of any factorization or predefined priors on the bases, we work directly with the shape $\mathbf{X}$, so we are estimating both coefficients and bases at the same time. For later computations, we define a new time-varying shape matrix $\mathbf{X}^{\#}$ that re-arrange the rows of $\mathbf{X}$ that correspond to X; Y ,and Z coordinate separately, in an $3N \times F$ matrix form as:

$$\mathbf{X}^{\#} = \begin{bmatrix} X_{11} & X_{12} & \cdots & X_{1f} \\ X_{21} & X_{22} & \cdots & X_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ X_{n1} & X_{n2} & \cdots & X_{nf} \\ Y_{11} & Y_{12} & \cdots & Y_{1f} \\ Y_{21} & Y_{22} & \cdots & Y_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ Y_{n1} & Y_{n2} & \cdots & Y_{nf} \\ Z_{11} & Z_{12} & \cdots & Z_{1f} \\ Z_{21} & Z_{22} & \cdots & Z_{2f} \\ \vdots & \vdots & \ddots & \vdots \\ Z_{n1} & Z_{n2} & \cdots & Z_{nf} \end{bmatrix} \in \mathbb{R}^{3N\times F}.$$

This approach proposes a simple method without making any prior assumption to the non-rigid shape other than having low-rank, producing a reliable result without suffering from inherent basis-ambiguities, which many conventional factorization methods are affected. To achieve this, the proposed problem is:

$$\begin{aligned} &\min_{\mathbf{X}^{\#}} \ \text{rank} \ (\mathbf{X}^{\#}) \\ &s.t. \ \mathbf{W} = \mathbf{R}\mathbf{X} \end{aligned} \tag{2.4}$$

The shape rank minimization problem is difficult to solve due to the discrete nature of the rank function. As a common practice in rank minimization problems, the rank function is relaxed with the nuclear/trace norm minimization. Resulting in the following optimization problem, which can be solved efficiently as:

$$\begin{aligned} &\min_{\mathbf{X}^{\#}} \ \|\mathbf{X}^{\#}\|_{*} \\ &s.t. \ \mathbf{W} = \mathbf{R}\mathbf{X} \end{aligned} \tag{2.5}$$

Note that Dai *et al.* [3] approach minimizes the nuclear norm of $\mathbf{X}^{\#} \in \mathbb{R}^{3N \times F}$, rather than $\mathbf{X} \in \mathbb{R}^{3F \times N}$. This is done because [3] noticed that $\mathbf{X}$ is not a fully-generic rank-3K matrix, since in reality there are in fact only K shape bases (rather than $3K$). Ignoring this structure will lead to higher degrees of freedom, i.e. more ambiguities, so re-arranging shape matrix $\mathbf{X}$ into $\mathbf{X}^{\#}$ is preferable as it captures the essence of the K-order linear combination model, attempting to directly learn redundancies between frames.

Despite the fact that [3] present a simple and elegant approach for 3D reconstruction, it is not a quite robust method when we are dealing with complex motions as we want to model in this work, since this approach implicitly assumes that the underlying data structure is a single low-rank subspace as it happens in Robust PCA [25]. Liu et al. [4] noticed that when data is drawn from a union of subspaces, like in complex motions, RPCA, and hence [3] approach, treat the data as a summation of subspaces $(\mathbf{S}_1, \mathbf{S}_2, \mathbf{S}_3, \ldots)$ rather than a union. Since the sum $\Sigma_{i=1}^{k} \mathbf{S}_i$ can be much larger than the union, the individual subspaces are not well considered and so the recovery may be inaccurate.

Knowing that fact one could thing that a good strategy would be to cluster the complex motion into individual subspaces before applying the method proposed by [3] to each of

them. Despite it seems a good strategy, the problems appear as soon as we try to cluster the motion only with the measurement matrix $\mathbf{W}$, i.e., using 2D motion instead of 3D. Apart from the fact that the number of clusters estimation is a quite complex problem, the camera motion and projection ambiguities lead to a poor clustering accuracy, which makes it difficult afterwards to reconstruct the shape properly. We will show results to validate this claim in the next section.

Chapter 3

Affinity Matrix and Clustering

Low-Rank Representation (LRR) is a method proposed by Liu *et al.* [4] which given a set of data samples approximately drawn from a union of multiple subspaces, aims to cluster those samples into their respective subspaces removing also possible outliers. The output result of this algorithm is an affinity matrix, named $\mathbf{Z} \in \mathbb{R}^{F \times F}$ where it is shown if each sample, i.e., the shape of the body for each frame, has a some kind of similarity with the other shapes in other frames.

The formulation used to this purpose is a generalization based on RPCA [25], to better handle the mixed data. It seeks the LRR method among all the possibilities that can represent the data samples as a linear combination of them. As in [3], the rank function minimization has been relaxed to the nuclear form minimization, such that:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_l \\ &s.t. \ \ \mathbf{X} = \mathbf{A}\mathbf{Z} + \mathbf{E} \end{aligned} \tag{3.1}$$

where $\mathbf{A}$ is a dictionary that linearly spans the data space and λ represents a weight. This equation can be seen as a bit tricky because the data matrix, which itself can contain errors, is used as the dictionary for error correction. However, in problems like ours is indeed a good choice [4], replacing the dictionary $\mathbf{A}$ by the shape $\mathbf{X}$:

$$\begin{aligned} &\min_{\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \lambda \|\mathbf{E}\|_l \\ &s.t. \ \ \mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E} \end{aligned} \tag{3.2}$$

In order to show how an affinity matrix looks like, and to exemplify how are the clustering results using the 2D projected information rather than the 3D shape, we have done a quick experiment using a sequence from the CMU MoCap database called *pickup*. In this sequence a person is being tracked using 41 points during 357 frames, while he/she is performing 3 different actions: stand, pick a box from the floor and give the box to another person by extending the arms.

The datasets provides the object shape in a 3D space, so in order to exemplify that clustering from the 2D projection is a bad idea, we have generated the measurement matrix $\mathbf{W}$ by an orthographic assumption $\mathbf{W} = \mathbf{RX}$, using as $\mathbf{R}$ a synthetic rotation matrix, simulating a 5 degree rotation per frame around the human body in the z-axis. In Fig. 3.1 we display a comparison between the 2D clustering and the original one done with the 3D ground truth shape, the results are quite far from being good.

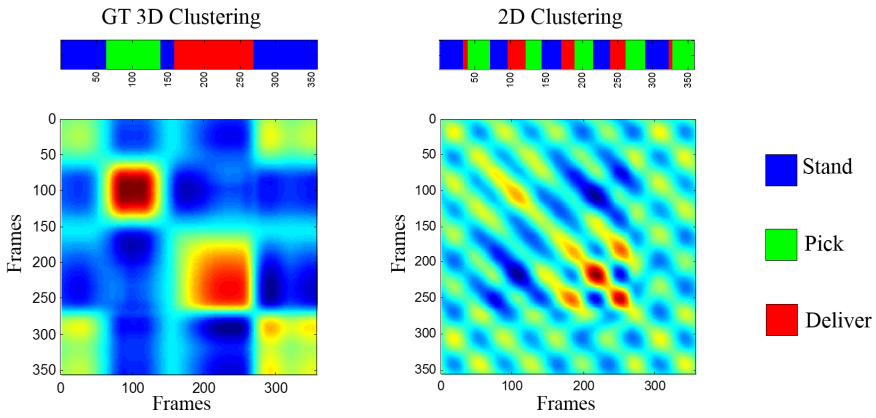

Figure 3.1: ***Affinity matrices and clustering using 3D and 2D information, respectively.*** *Comparison between results obtained applying LRR to the ground truth 3D shape from the* pickup *sequence and LRR to its 2D synthetic projection created. While the clustering can be learned from data using 3D information, we can observe bad results when 2D information is used.*

The results obtained for the ground truth are obviously satisfying, showing clearly in the affinity matrix 3 groups of frames that seem to encode similar shapes (first and last frames together, and two more groups in the center which diverge from the rest of the sequence), making a direct correspondence to the actions performed in during the sequence. If we apply a K-means clustering operation to the affinity matrix [26], the ”clustering bar” on the top is obtained, which explains clearly the affinity matrix encoding in 3 different colors/clusters. In color blue is encoded the stand up position, the green shows the part when the person blend over the hips to pick up the box from the floor, and in red when the person delivers it.

On the other hand, the results obtained with 2D clustering, are far from being correct. The

affinity matrix does not show the sharp edges as before, being a completely chaos without any clear similarities between frames, probably due to the camera rotation together with the projection ambiguities.

Chapter 4

Joint Clustering and Reconstruction Estimation

In this section is presented the algorithm combining both approaches in a single one, attempting to solve both tasks simultaneously. The subspace clustering constraint $\mathbf{X} = \mathbf{XZ} + \mathbf{E}$ taken from [4] enforces the union of subspaces structure of $\mathbf{X}$ in a low-rank coefficients matrix $\mathbf{Z}$, while the NRSfM constraint $\mathbf{W} = \mathbf{RX}^{\#1}$ from [3] continually performs the 3D reconstruction and registration from the 2D measurement matrix $\mathbf{W}$, into the 3D shape matrix $\mathbf{X}$ in each one of the subspaces created by the first constraint. Finally, our problem is:

$$\begin{aligned} &\min_{\mathbf{X},\mathbf{Z},\mathbf{E}} \ \|\mathbf{Z}\|_* + \gamma\|\mathbf{X}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\ &s.t. \ \ \mathbf{X} = \mathbf{XZ} + \mathbf{E} \\ &\qquad \mathbf{W} = \mathbf{RX}^{\#} \end{aligned} \tag{4.1}$$

where γ and λ are penalty parameters for $\|\mathbf{X}\|_*$ and $\|\mathbf{E}\|_{2,1}$, respectively. The $l_{2,1}$ norm has been chosen to characterize the error term $\mathbf{E}$ since we want to model the sample-specific corruptions and outliers. This norm is a good relaxation of the $l_{2,0}$ norm adopted by [4].

The constrained convex problem in Eq. (4.1) can be solved by various methods. For efficiency we have optimized it using Augmented Lagrangian Multiplier (ALM) method [27].

[1] For simplicity, in this chapter we represent the projection equation by means of $\mathbf{X}^{\#}$, a $3F \times N$ matrix, in contrast to $\mathbf{X}$ that is used in the clustering constraint and corresponds to a $3N \times F$ matrix. Both matrices contain the same information, the 3D shape scene.

But first we convert (4.1) to the following equivalent problem:

$$\begin{aligned} &\min_{\mathbf{X},\mathbf{Z},\mathbf{E},\mathbf{H},\mathbf{J}} \|\mathbf{J}\|_* + \gamma\|\mathbf{X}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\ s.t.\ \ &\mathbf{X} = \mathbf{X}\mathbf{Z} + \mathbf{E} \\ &\mathbf{W} = \mathbf{R}\ \mathbf{H} \\ &f(\mathbf{X}) = \mathbf{H} \\ &\mathbf{Z} = \mathbf{J} \end{aligned} \tag{4.2}$$

where we impose two new constraints: $\mathbf{Z} = \mathbf{J}$ which splits the trace minimization of $\mathbf{Z}$ in two steps, and $f(\mathbf{X}) = \mathbf{H}$ which establish a connection between matrices $\mathbf{X} \equiv \mathbf{H}$ and $\mathbf{X}^{\#}$, both containing the same elements but with different organization (the explanation of this distinction has been elucidated in section 2.2). For simplicity, we define a function f that rearranges a matrix $\mathbf{A} \in \mathbb{R}^{3N\times F}$ into $\mathbf{A}^{\#} \in \mathbb{R}^{3F\times N}$, being f^{-1} its opposite operation. The compact form of these functions are expressed in 4.3 and 4.4, respectively, even though both have been omitted during the formulation to simplify it. With $\mathbf{T} \in \mathbb{R}^{9N\times N}$ and $\mathbf{P} \in \mathbb{R}^{9F\times F}$ being some properly defined 0/1 matrices allowing the re-arrangement, similar to permutation matrices.

$$\mathbf{A}^{\#} = f(\mathbf{A}) \ \rightarrow \ \mathbf{A}^{\#} = (\mathbf{A}^T \otimes \mathbf{I}_3)\,\mathbf{P} \tag{4.3}$$

$$\mathbf{A} = f^{-1}(\mathbf{A}^{\#}) \ \rightarrow \ \mathbf{A} = (\mathbf{I}_3 \otimes (\mathbf{A}^{\#})^{\top})\,\mathbf{T} \tag{4.4}$$

Equation (4.2) can be solved with ALM method, which minimizes the following Augmented Lagrangian cost function:

$$\begin{aligned} &\min_{\mathbf{X},\mathbf{Z},\mathbf{E},\mathbf{H},\mathbf{J}} \mathcal{L} = \|\mathbf{J}\|_* + \gamma\|\mathbf{X}\|_* + \lambda\|\mathbf{E}\|_{2,1} \\ &+ < \mathbf{\Gamma}_1, \mathbf{X} - \mathbf{X}\mathbf{Z} - \mathbf{E} > + \frac{\mu}{2}\|\mathbf{X} - \mathbf{X}\mathbf{Z} - \mathbf{E}\|_F^2 \\ &+ < \mathbf{\Gamma}_2, \mathbf{W} - \mathbf{R}\mathbf{H} > + \frac{\mu}{2}\|\mathbf{W} - \mathbf{R}\mathbf{H}\|_F^2 \\ &+ < \mathbf{\Gamma}_3, f(\mathbf{X}) - \mathbf{H} > + \frac{\mu}{2}\|f(\mathbf{X}) - \mathbf{H}\|_F^2 \\ &+ < \mathbf{\Gamma}_4, \mathbf{Z} - \mathbf{J} > + \frac{\mu}{2}\|\mathbf{Z} - \mathbf{J}\|_F^2 \end{aligned} \tag{4.5}$$

4.1 Algorithm

Now the problem is unconstrained so it can be minimized with respect to $\mathbf{X}, \mathbf{Z}, \mathbf{E}, \mathbf{H}$ and $\mathbf{J}$, by fixing the others respectively, and updating multipliers $\mathbf{\Gamma}_1 \in \mathbb{R}^{3N\times F}$, $\mathbf{\Gamma}_2 \in \mathbb{R}^{2F\times N}$, $\mathbf{\Gamma}_3 \in \mathbb{R}^{3N\times F}$ and $\mathbf{\Gamma}_4 \in \mathbb{R}^{F\times F}$, together with the penalty parameter $\mu > 0$ after each iteration. Algorithm 1 shows the steps followed in each iteration clearly. In case of estimating rotation matrix $\mathbf{R}$ from the other measurement matrix $\mathbf{W}$ input, the process has to be done previously to execution of the algorithm proposed by [3], that gives a simple approach to $\mathbf{R}$ estimation, however, during our experiments we use ground truth $\mathbf{R}$ all the time. The outline of the algorithm is shown in algorithm 1.

Note that although steps belonging to variables $\mathbf{J}, \mathbf{X}$ and $\mathbf{E}$ are convex problems, all them have closed-from solutions. Steps 1 and 3 are solved via the Singular Value Thresholding (SVT) operator [28], while step 5 is solved via lemma 1 proposed in [29]. A maximum number of iteration could be defined also, in case the convergence conditions are not really accomplished in a significant amount of iterations.

Lemma 1 *Let* $\mathbf{Q}$ *be a given matrix. If the optimal solution to:*

$$\min_{\mathbf{E}} \alpha\|\mathbf{E}\|_{2,1} + \frac{1}{2}\|\mathbf{E}-\mathbf{Q}\|_F^2$$

is $\mathbf{E}^*$*, then the ith column of* $\mathbf{E}^*$ *is:*

$$[\mathbf{E}^*]_{:,i} = \begin{cases} \frac{\|[\mathbf{Q}]_{:,i}\|_2-\alpha}{\|[\mathbf{Q}]_{:,i}\|_2}, & if \|[\mathbf{Q}]_{:,i}\|_2 > 0 \\ 0 \quad , & otherwise \end{cases}$$

Algorithm 1 Minimizing energy $\mathcal{L}$ in Eq. (4.5)

Input: data matrix $\mathbf{W}$, rotation matrix $\mathbf{R}$, and penalty parameters γ and λ.

Initialize: $\mathbf{X} = \mathbf{Z} = \mathbf{H} = \mathbf{E} = 0$, $\mathbf{\Gamma}_1 = \mathbf{\Gamma}_2 = \mathbf{\Gamma}_3 = \mathbf{\Gamma}_4 = 0$, $\rho = 1.1$, $\mu = 10^{-2}$, $\mu_{max} = 10^6, \epsilon = 10^{-7}$

while not converged **do**

1. Fix the others and update $\mathbf{J}$ by

$$\mathbf{J} = \min \tfrac{1}{\mu}\|\mathbf{J}\|_* + \tfrac{1}{2}\|\mathbf{J} - (\mathbf{Z} + \tfrac{\mathbf{\Gamma}_4}{\mu})\|_F^2$$

2. Fix the others and update $\mathbf{Z}$ by

$$\mathbf{Z} = (\mathbf{X}^\top\mathbf{X} + \mathbf{I}_F)^{-1}(\mathbf{X}^\top(\mathbf{X} - \mathbf{E}) + \mathbf{J} + (\mathbf{X}^\top\mathbf{\Gamma}_1 - \mathbf{\Gamma}_4)/\mu)$$

3. Fix the others and update $\mathbf{X}$ by

$$\mathbf{X} = \min \tfrac{\gamma}{\mu}\|\mathbf{X}\|_* + \tfrac{1}{2}\|\mathbf{X} - (f^{-1}(\mathbf{H}) + (\mathbf{E} - \tfrac{\mathbf{\Gamma}_1}{\mu})\mathbf{M}^T - f^{-1}(\tfrac{\mathbf{\Gamma}_3}{\mu}))(\mathbf{M}\mathbf{M}^T + \mathbf{I}_F)^{-1}\|_F^2$$

where: $\mathbf{M} = (\mathbf{I}_F - \mathbf{Z})$

4. Fix the others and update $\mathbf{H}$ by

$$\mathbf{H} = (\mathbf{R}^\top\mathbf{R} + \mathbf{I}_{3F})^{-1}(\mathbf{R}^\top(\mathbf{W} + \tfrac{\mathbf{\Gamma}_2}{\mu}) + f(\mathbf{X}) + \tfrac{\mathbf{\Gamma}_3}{\mu})$$

5. Fix the others and update $\mathbf{E}$ by

$$\mathbf{E} = \min \tfrac{\lambda}{\mu}\|\mathbf{E}\|_{2,1} + \tfrac{1}{2}\|\mathbf{E} - (\mathbf{X} - \mathbf{X}\mathbf{Z} + \tfrac{\mathbf{\Gamma}_1}{\mu})\|_F^2$$

6. Update multipliers by

$$\mathbf{\Gamma}_1^{k+1} = \mathbf{\Gamma}_1^k + \mu(\mathbf{X} - \mathbf{X}\mathbf{Z} - \mathbf{E})$$
$$\mathbf{\Gamma}_2^{k+1} = \mathbf{\Gamma}_2^k + \mu(\mathbf{W} - \mathbf{R}\mathbf{H})$$
$$\mathbf{\Gamma}_3^{k+1} = \mathbf{\Gamma}_3^k + \mu(f(\mathbf{X}) - \mathbf{H})$$
$$\mathbf{\Gamma}_4^{k+1} = \mathbf{\Gamma}_3^k + \mu(\mathbf{Z} - \mathbf{J})$$

7. Update parameter μ by

$$\mu = \min(\rho\mu, \mu_{max})$$

8. Check the convergence conditions

$$\|\mathbf{X} - \mathbf{X}\mathbf{Z} - \mathbf{E}\|_\infty < \epsilon \text{ and } \|\mathbf{W} - \mathbf{R}\mathbf{H}\|_\infty < \epsilon \text{ and}$$
$$\|f(\mathbf{X}) - \mathbf{H}\|_\infty < \epsilon \text{ and } \|\mathbf{Z} - \mathbf{J}\|_\infty < \epsilon$$

end while

Chapter 5

Experimental Results

In order to evaluate the performance of our algorithm, which attempts to simultaneously recover the 3D shape and the affinity matrix, we selected various sequences from two different databases, the CMU Motion Capture (MoCap) and the Utrecht Multi-Person Motion (UMPM). Both of them provide sequences involving people performing different actions while interacting with objects the environment. In all of them, the 3D shape is given by means of the joint sensors the subject is wearing, so in that way we have the 3D ground truth to compare against the results obtained.

In each sequence tested, a synthetic camera rotating around the human body is applied in order to obtain the measurement matrix $\mathbf{W}$, by means of an orthographic projection. The camera is always pointing to the center of the moving object, with a relative speed with respect to the subject of 5 degrees per frame. This rotation $\mathbf{R}$ matrix is the one used as a input in our algorithm for all tests, leaving the performance evaluation while estimation it for a future work.

In the case the subject from any of the sequences performs any rotation while walking or moving, the sequence will be divided taking only the parts where there are no changes. Otherwise, the synthetic rotation applied overlaps the natural subject rotation, creating a confusion in the algorithm. If it is the case, it will be specified the portion taken in frames.

So, for measuring how well our approach performs, during all the experiments we have evaluated basically two main metrics, the accuracy in the subspace clustering and the accuracy in the 3D reconstruction. For quantitative evaluation, we will follow the metrics already

used in [3, 21], and will report the normalized mean 3D error e_S, defined as:

$$e_S = \frac{1}{\sigma FN}\sum_{f=1}^{F}\sum_{n=1}^{N} e_n^f, \quad \sigma = \frac{1}{3F}\sum_{t=1}^{F}(\sigma_x^f + \sigma_y^f + \sigma_z^t),$$

where e_n^f is the 3D reconstruction error for the n-th point at frame f. σ_x^f, σ_y^f and σ_z^f indicate the standard deviations at frame f of the x-, y- and z-coordinates of the original shape.

5.1 Clustering performance

Subspace clustering splits the frames in a sequence in a series of subsets/groups where the non-rigid shapes in each of them takes part from the same subspace. So in order to test how accurate is our clustering, each frame is compared against the clustering results using the ground truth 3D shape, checking whether the same frame belongs to the cluster/subspace or not. Obtaining at the end a success ratio in percentage, being 1 if all the frames coincide.

The number of clusters picked in each sequence for doing the clustering varies. In some of them the number is quite clear because the actions are well differentiated one from the others, but there are others where can be confusing, so we picked finally the amount of clusters giving the best results, but always taking into account the primitive motions inside the sequence. This choice does not affect by any means to 3D recovering nor the affinity matrix obtainment, because the clustering step is applied afterwards.

The sequences picked for clustering accuracy evaluation are: ”stretch”, ”yoga”, ”pickup”, ”drink”, sequences 5th, 9th, 10th,11th,13th,14th,15th from subject 86 from CMU MoCap database, and ”p1_ortho_2” from UMPM. Some of them are split up for containing natural rotation of the subject, the frames selected in that case are specified in table 5.1.

Notice that while in some sequences the accuracy achieved is above 95%, in some others, like the ones from subject 86 to be more precise, the performance on the accuracy goes down until until around 70%. In order to understand the reason beneath that results we have taken the sequence with lowest result, which is the ”86_11”,and we have compared it against the ”86_09”, which belongs to the same subject 86 and contains a similar number of frames. Affinity matrices and clustering bars from both are obtained (see Fig. 5.1 and 5.2), retrieving also their respective ground truths by applying LRR to the original and known 3D shape.

One quickly realize while in sequence ”86_09” (Fig. 5.1) the primitives are clearly sep-

Sequence	Frames	Cluster Accuracy	# Clusters
Stretch	All	0.945	3
Yoga	All	0.974	2
Pickup	All	0.995	2
Drink	All	0.963	2
86_05	200-650	0.797	3
86_09	220-720	0.976	4
86_10	400-800	0.827	3
86_11	200-650	0.687	3
86_13	220-1050	0.726	4
86_14	220-500	0.801	2
86_15	300-1300	0.707	3
p1_orthosyn_2	All	0.982	2

Table 5.1: Clustering accuracy over 11 sequences involving different primitives.

arated 4 main blocks during time, the actions in ”86_11” sequence are mixed quite fast, jumping from one primitive to another quite fast. This leads to more inaccuracies while predicting which frame belong to which subspace, since the reconstructed affinity matrix is not as precise as it could be the original.

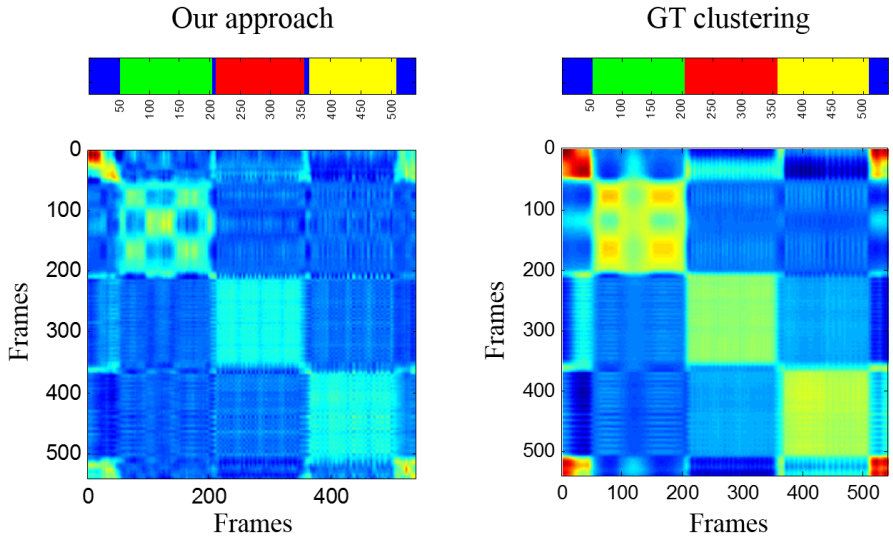

Figure 5.1: Affinity matrix and clustering bar on the sequence 86_09 from frame 220 to 720. The sequence includes 4 actions or primitives: stand (blue), sitting with the hand in the forehead (green), sitting while clapping(red), and stand while clapping (yellow). Our methods gets similar results when comparing to the ground truth, obtaining an accuracy up to 97.6%.

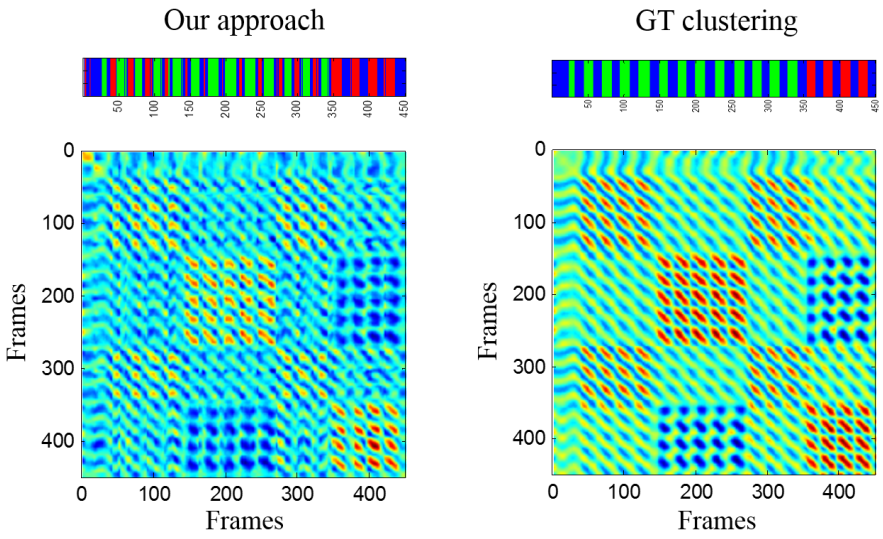

Figure 5.2: Affinity matrix and clustering bar on the sequence 86_11 from frame 200 to 650. The sequence includes 3 actions o primitives: stand (blue), one arm up (green) and two arms up(red). In that case our methods suffers in the reconstruction due to the quick transition between primitives.

5.2 3D Reconstruction Accuracy

The second part of the testing consists in measuring the 3D reconstruction accuracy from 2D projection in complex non-rigid motions, which is the main goal of our algorithm. For this purpose, we use the same sequences as in the clustering accuracy evaluation, always applying the synthetic rotation matrix with 5 degrees per frames to obtain the 2D projection. Those sequence contain multiple human actions and object interaction, so we can consider them complex motions, suitable for our purpose. All the reconstruction values will be compared against [3], which is state-of-art in the field. The results can be seen in 5.2.

As expected, the error obtained when performing the non-rigid reconstruction with our method is lower than with [3] counterpart. As we are dealing with complex movement involving different primitives, our algorithm is able to cluster each of them into their respective subspace, while [3] works from a summation of subspaces, leading to more inaccuracies.

Although the results are satisfactory in most of the sequences tested, there are some like sequence number 5 from the subject 86 (”86_05”), which present a high degree of error compared with the rest. We believe this is a consequence of not having enough frames inside the clusters to perform correctly the low-rank conditions that allows to minimize the error.

To illustrate better the performance of our method, we have chosen 4 different sequences from the ones used in during the experiments, showing a wide range of activities done by a person implying different kind of deformations in the legs, torso and arms. The sequences

Sequence	Frames	Our approach	SPM [3]
Stretch	All	0.036	0.050(8)
Yoga	All	0.028	0.034(9)
Pickup	All	0.035	0.056(7)
Drink	All	0.020	0.038(4)
86_05	200-650	0.256	0.363(8)
86_09	220-720	0.064	0.090(9)
86_10	400-800	0.089	0.096(11)
86_11	200-650	0.119	0.1312(7)
86_13	220-1050	0.102	0.140(7)
86_14	220-500	0.154	0.162(9)
86_15	300-1300	0.078	0.113(7)
p1_orthosyn_2	All	0.137	0.155(7)

Table 5.2: Reconstruction accuracy over 11 sequences involving different primitives. For the SPM [3] baseline, we also represent in brackets the rank of the shape basis that was used to obtain the results.

picked are: ”86_09”,”86_09”,”86_09” and ”p1_orthosyn”, the first three coming from CMU MoCap database, and the last from UMPM. The figures showing some reconstructed frame in each sequence are (5.3),(5.4), (5.5) (5.6) respectively. In all of them the blue skeleton represents our 3D estimation and the red circles are the 3D ground truth shape, the closer blue dots to red circles the best.

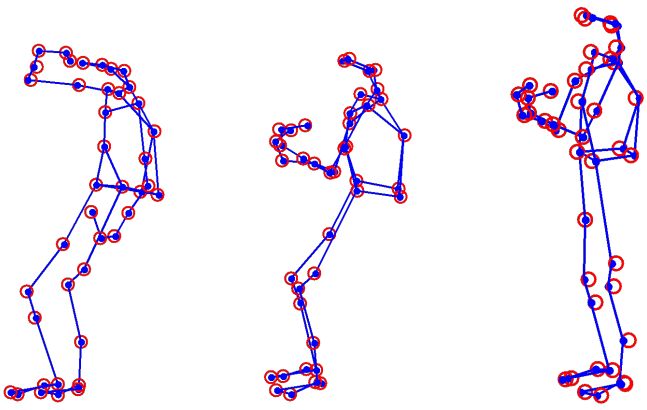

Figure 5.3: Sequence ”86_13” includes 4 actions or primitives: stand, sitting with the hand in the forehead, sitting while clapping, and stand while clapping. Except the stand position the resting 3 can be seen in the figure in the same order described. The error achieved in this sequence has been very low, hence the high proximity between the blue dots and the red.

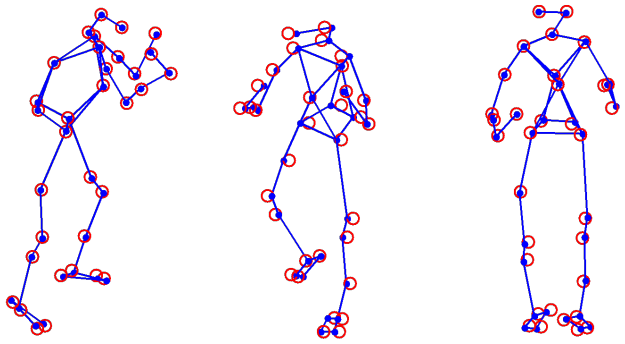

Figure 5.4: Sequence "86_13" shows a person climbing and going down a ladder. In the first frame the person raises the left foot to the first step of the ladder, then the subject continues climbing until the top, shown in the third frame.

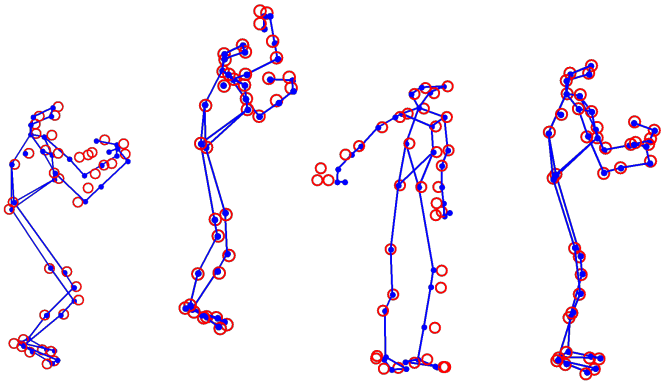

Figure 5.5: In sequence "86_14" appears a subject playing basketball. On the frames showed is captured one of the many shoots done along the sequence.

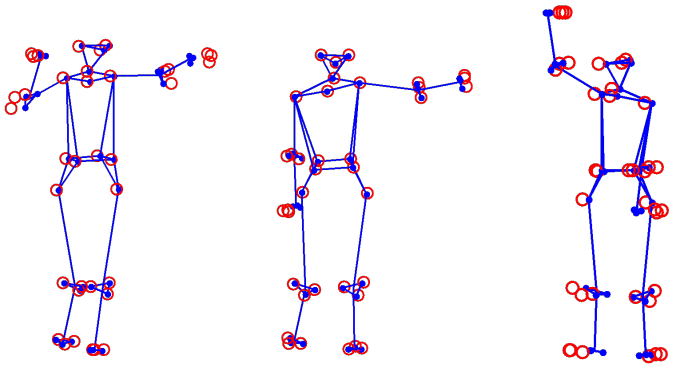

Figure 5.6: Sequence "p1_othosyn" includes a human body moving laterally while positioning the arms like doing some kind of signals. In the figure are shown 3 different frames, each one with a distinct gesture.

Chapter 6

Temporal Planning and Costs

In this Chapter, the definition of tasks and their temporal scheduling plan are defined.

6.1 Tasks identification

The project is divided into a set of tasks (iterations) each one divided in analysis, implementation, testing and documentation. The application of agile methodologies will allow a more close up feedback relationship with the project supervisors as well as a faster response against obstacles and incidents.

6.1.1 State of the art

In this first task an evaluation of the current state of the art will be analyzed. The main objective of this task is to identify the most promising state-of-the-art non-rigid structure reconstruction algorithms, as well as subspace clustering techniques.

6.1.2 Learning

Once identified the proper algorithms, study all the necessary theory in order to understand and be able to implement the method.

6.1.3 Implementation and debbuging

Implementation of the algorithm in the Matlab environment, and debug to make some correction in previous steps in case the method does not work properly as expected.

6.1.4 Tunning and optimization

Once our approach is fully functional is time for tunning the parameters of our algorithm in order to achieve the best results with various type sequences.

6.1.5 Experimental results

The final steps before the documentation is to validate the approach with different experiments evaluating the performance in various aspects. In this work we are trying to reconstruct 3D non-rigid structures, while clustering the motion in subspaces, so during the results we need to focus in that issues.

6.1.6 Final documentation

At each iteration the documentation of that task will be generated. Hence, in this final iteration only final polishes over the documentation will be made.

6.2 Time Planning

The total period developing the project has been almost 6 months. However, during the first 5 months, from April to August, I have been doing a full time internship, which unfortunately reduced the available amount of time to devote in the project. The main part of the work, i.e. task involving tunning and optimization and doing all the experimentation, has been done in the last month and a half, from mid August to October. While the state-of-the-art search and learning together with the implementation and debugging was done during the initial months, taking more time than planned at the beginning. In table 6.1 appears the final time devoted to each one of the tasks.

Task	Time [h]
State of the art	50
Learning	60
Implementation and debugging	210
Tunning and Optimization	110
Experimental Results	115
Final documentation	55
TOTAL	**600**

Table 6.1: Time planning table of all task involved for the project development.

6.3 Costs

To fulfill this project a set of hardware and software resources are needed. In this chapter a cost analysis is presented. For hardware and software costs have been computed the amortization during the expected service lifetime. And we want to add also overhead costs, expenses which are not directly related with the project, but allow the proper development. The total cost calculated over the 6 months is 3613 €.

$$Total\ costs = HW + SW + overhead = 108 + 85 + 3420 = 3613 Euros$$

6.3.1 Hardware costs

Product	Price	Units	Lifetime	Amortization
Sony VAIO	1080 €	1	5 years	18 €
TOTAL	**1080 €**	-	-	**108 €**

Table 6.2: Hardware costs during the project development.

6.3.2 Software costs

Product	Price	Units	Lifetime	Amortization
Matlab Student	70 €	1	1 year	35 €
Matlab Add-ons	100 €	1	1 year	50 €
ShareLatex	0	1	1 year	-
TeXstudio	0	1	1 year	-
TOTAL	**170 €**	-	-	**85 €**

Table 6.3: Software costs during the project development.

6.3.3 Overhead costs

Concept	Price	Time	Total
Rent and utilities	500 €/month	6 months	3000 €
Transport	70 €/month	6 months	420 €
TOTAL	**570** €/month	-	**3420** €

Table 6.4: Overhead costs related to the project.

Chapter 7

Conclusions

In this project, we have presented a novel approach of recovering the 3D shape of a non-rigid object using a single algorithm, which combines the better of low-rank shape models along with the spectral clustering by using LRR. In this manner we are forcing the complex motion to adhere to a union of subspaces while the shape reconstruction is done in each one of the subspaces as it was a single one.

To prove the benefits of this combined approach some experiments are done with complex motions, i.e., sequences including many temporal primitive actions. The results obtained have evaluated from two different points of view, the 3D shape estimation accuracy and the clustering accuracy, all of them from 2D point tracks in a monocular image sequence.

The shape estimation seems promising, overpassing state-of-art methods in the terms of accuracy, under the same priors. However, if there is lack of frames in the subspace clustering, the reconstruction could be affected because there are not enough data for performing the low-rank condition. Despite this fact, which need to be studied in detail, the results of this new approach are good enough.

Regarding the clustering exactness, we have seen that the higher or lower grade of accuracy obtained depends on how the primitive actions are executed within the sequence. It is preferable if the primitive are changed in a smooth manner rather than a quick alternating, to obtain better results. Nonetheless, the clustering does not affect direct manner to the shape estimation being able to obtain acceptable 3D reconstructions with quite poor clustering.

7.1 Future Work

This work has had a rather qualitative approach, and the goal has been to study and introduce an effective method for the aforementioned problem. Further steps could include:

- Based on the clustering learned from data, we could automatically classify activities from monocular video.
- Automatic feature detection tracking to compute the observation matrix on the fly.

Bibliography

[1] Life Sciences. Vicon motion services. In https://www.vicon.com/motion-capture/life-sciences, 2016.

[2] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[3] Y. Dai, H. Li, and M. He. A simple prior-free method for non-rigid structure from motion factorization. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2018–2025, 2012.

[4] S. Yan J. Sun Y. Yu G. Liu, Z. KLin and Y. Ma. Robust recovery of subpaces structure by low-rank representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):171–184, 2013.

[5] Y. Zhu, D. Huang, F. De La Torre, and S. Lucey. Complex non-rigid motion 3D reconstruction by union of subspaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1542–1549, 2014.

[6] C. Bregler, A. Hertzmann, and H. Biermann. Recovering non-rigid 3D shape from image streams. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 690–696, 2000.

[7] C. Tomasi and T. Kanade. Shape and motion from image streams under orthography: A factorization approach. *International Journal of Computer Vision*, 9(2):137–154, 1992.

[8] I. Akhter, Y. Sheikh, and S. Khan. In defense of orthonormality constraints for nonrigid structure from motion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1534–1541, 2009.

[9] V. Blanz and T. Vetter. A morphable model for the synthesis of 3D faces. In *International Conference and Exhibition on Computer Graphics and Interactive Techniques*, pages 187–194, 1999.

[10] F. Moreno-Noguer and J. M. Porta. Probabilistic simultaneous pose and non-rigid shape recovery. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1289–1296, 2011.

[11] J. Barbic and D. James. Real-time subspace integration for st. venant-kirchhoff deformable models. *ACM Transactions on Graphics*, 24(3):982–990, 2005.

[12] A. Agudo, J. M. M. Montiel, L. Agapito, and B. Calvo. Modal space: A physics-based model for sequential estimation of time-varying shape from monocular video. *JMIV, to appear*, 2016.

[13] A. Agudo, J. M. M. Montiel, B. Calvo, and F. Moreno-Noguer. Mode-shape interpretation: Re-thinking modal space for recovering deformable shapes. In *IEEE Winter Conference on Applications of Computer Vision*, pages 1–8, 2016.

[14] A. Del Bue, X. Llado, and L. Agapito. Non-rigid metric shape and motion recovery from uncalibrated images using priors. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1191–1198, 2006.

[15] A. Agudo and F. Moreno-Noguer. Recovering pose and 3D deformable shape from multi-instance image ensemble. In *Asian Conference on Computer Vision*, 2016.

[16] L. Torresani, A. Hertzmann, and C. Bregler. Nonrigid structure-from-motion: estimating shape and motion with hierarchical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(5):878–892, 2008.

[17] A. Agudo, L. Agapito, B. Calvo, and J. M. M. Montiel. Good vibrations: A modal analysis approach for sequential non-rigid structure from motion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1558–1565, 2014.

[18] A. Agudo and F. Moreno-Noguer. Simultaneous pose and non-rigid shape with particle dynamics. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2179–2187, 2015.

[19] A. Bartoli, V. Gay-Bellile, U. Castellani, J. Peyras, S. Olsen, and P. Sayd. Coarse-to-fine low-rank structure-from-motion. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2008.

[20] I. Akhter, Y. Sheikh, S. Khan, and T. Kanade. Trajectory space: A dual representation for nonrigid structure from motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(7):1442–1456, 2011.

[21] P. F. U. Gotardo and A. M. Martinez. Non-rigid structure from motion with complementary rank-3 spaces. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3065–3072, 2011.

[22] A. Agudo, F. Moreno-Noguer, B. Calvo, and J. M. M. Montiel. Sequential non-rigid structure from motion using physical priors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(5):979–994, 2016.

[23] R. Garg, A. Roussos, and L. Agapito. Dense variational reconstruction of non-rigid surfaces from monocular video. In *CVPR*, 2013.

[24] A. Agudo and F. Moreno-Noguer. Learning shape, motion and elastic models in force space. In *IEEE International Conference on Computer Vision*, pages 756–764, 2015.

[25] E. Cands, X. LI, Y. Ma, and J. Wright. Robust principal component analysis? *J. ACM*, 58(3):1–37, 2009.

[26] U. von Luxburg. A tutorial on spectral clustering. *Statistics and Computing*, 17, 2007.

[27] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optmization and statistical learning via the alternated direction method of multipliers. *Found. Trends Machine Learning*, 3(1), 2011.

[28] J. Cai, E. Cands, and Z. Shen. A singular value thresholding algorithm for matrix completion. *SIAM J. Optimization*, 20(4):1956–1982, 2010.

[29] J. Yang, W. Yin, Y. Zhang, and Y. Wang. A fast algorithm for edge-perserving variational multichannel image restoration. *SIAM J. Imaging Sciences*, 2(2):569–592, 2009.