

# A Stereo Vision Approach for Cooperative Robotic Movement Therapy

Benjamin Busam<sup>1,2</sup>, Marco Esposito<sup>1</sup>, Simon Che'Rose<sup>2</sup>, Nassir Navab<sup>1,3</sup>, Benjamin Frisch<sup>1</sup>

<sup>1</sup> Computer Aided Medical Procedures, Technische Universität München, Germany

<sup>2</sup> FRAMOS Imaging Systems, Germany

<sup>3</sup> Computer Aided Medical Procedures, Johns Hopkins University, US

{b.busam, s.cherose}@framos.com, {marco.esposito, benjamin.frisch}@tum.de, navab@cs.tum.edu

## Abstract

*Movement therapy is an integrating part of stroke rehabilitation. The positive influence of intensive, repetitive motion training and the importance of active patient participation trigger the development of cooperative robotic assistants. We suggest a device for the re-education of upper limb movements in hemiparetic patients where a light-weight robotic arm that supports the deficient arm is equipped with a stereoscopic camera system. It follows the movements of the healthy arm that wears a sleeve equipped with flat round reflective markers detected by the cameras. We introduce an advanced robust and real-time algorithm to provide the tracking information. It performs a sparse marker based point cloud registration based on subpixel precision contour fits to enable high accuracy pose estimates while being capable of online model adjustments. The update rate of the tracking is 9 ms and the precision of the system is measured to be 0.5 mm. Tests with healthy subjects show that the system is able to accurately reproduce the movement of the healthy arm on an impaired arm.*

## 1. Introduction

Stroke affected 17 million people in 2010. At the same time, around 33 million stroke survivors were alive [6]. Considering that 80 % of these patients are affected by some kind of hemiparesis, i.e. 1-sided muscle weakness of the upper and/or lower limbs, there is a clear need for strategies that support their rehabilitation towards increasing their quality of life [19]. Upper limb therapy consists in a wide variety of approaches summarized by [19] and [1], that include neurological therapeutic approaches, electrical stimulation, mental practice and imagery, constraint-induced movement therapy, mirror therapy, and repetitive task practice. Due to their nature, the latter two are suited

for robotic support. While robot-assisted upper-limb rehabilitation indicates that intensive motion training is beneficial to the therapeutic outcomes both immediately after the stroke and in the longer term, patient participation turns out to be particularly important [2].

As mentioned by [1], implementations of robot-mediated therapy can be classified according to the type of human-robot interaction. We will focus on passive-mirrored systems, where the patient uses his unimpaired arm to control the robot that trains the affected arm. This review mentions the Bi-Manu-Track, MIME and BFIAMT, that are controlled via joysticks, and the U-EX07, that requires wearing an exoskeleton. To our best knowledge, there is no passive-mirrored upper limb rehabilitation system that would benefit from advances in computer vision to determine the movement of the guiding arm and use that information for real-time movement of the impaired arm.

This work is a first step towards contactless human robot interaction for upper limb rehabilitation where, as shown in fig. 1, a light-weight robotic arm is equipped with camera-in-hand visual servoing to mirror the natural movement of the healthy arm for the training of the impaired arm in hemiparetic patients. A major challenge is the robust and real-time tracking of the healthy arm required for an accurate reproduction of the hand movement by the robot. As such, this work particularly benefits from a marker-based tracking algorithm that uses circular marker shapes to fit geometric primitives on a subpixel precise level to their contours and consecutively performs a robust sparse 3D point cloud registration to a taught model cloud of the triangulated centres in real-time with submillimeter accuracy. The algorithmic pipeline also provides live parameter updates for the tracked upper limb to follow the arm even in case of natural skin deformations.

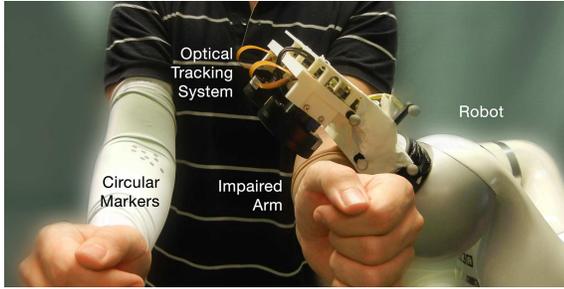


Figure 1. Photography of the system

## 2. Materials and Methods

### 2.1. Setup

As drawn in fig.2, we use a light-weight robotic arm LBR 4+ (KUKA, Germany) equipped with a custom-made 3D-printed mount that holds a stereoscopic camera system and with a sling that holds the impaired arm. The transformation between the camera coordinate system and the robot end effector is found with a hand-eye calibration procedure based on the Tsai-Lenz algorithm [27] in the eye-on-hand variant, with the implementation provided by the ViSP library [15].

The camera system consists of two GC1291M-BL board-level cameras (SMARTEK Vision, Croatia) with DSL315B-NIR miniature fisheye lenses (Sunex, USA) that allow for a horizontal field of view of 135°. Two IF 093 NIR band-pass filters (Schneider-Kreuznach, Germany) are used for both cameras. The FLDR-i70A direct ringlight illuminations (FALCON Illumination, Malaysia) flash at 875 nm wavelength and are triggered by an IPSC2 strobe controller (SMARTEK Vision, Croatia) during camera exposure with 750 mA at 24 VDC. Both cameras acquire synchronized images in hardware trigger mode with an exposure time of 1.5 ms at their maximum frame rate of 24 Hz. The camera parameters are estimated from a one-time calibration based on [31] that assumes the standard pinhole camera model with two radial distortion coefficients.

The healthy arm wears an immovable sleeve with flat circular markers with a diameter of 7 mm, made of retroreflective film (3M, USA). As illustrated in figure 3, these markers reflect infrared light emitted by a ring of LEDs around the lenses of the cameras. Due to the microstructure of the markers, most light is reflected back to its source - and thus the camera sensors - with minimum diffuse reflection in other directions. The band-pass filter on the sensor ensures high intensity peaks for the light reflected by the markers while suppressing noisy light sources. It facilitates the localization of the projected circles in the stereo image pair, explained in section 2.2 and makes it more robust against other circular structures within the visible area. The tracking algorithm, detailed in 2.3, extracts the relative position

and movement of the healthy arm with respect to the robot and provides the positioning information required to move the robot, and thus the impaired arm, using the principles introduced in section 2.4.

Image processing and tracking run on an Intel Core i7 960 at 3.2 GHz equipped with the FRAMOS Application Framework (FRAMOS Imaging Systems, Germany) while the robot is controlled by an Intel Core i5 4690K at 3.5 GHz. An external optical tracking system Polaris Vicra (NDI, Canada) is used to provide the ground truth for accuracy measurements. Latency tests are performed with a robot UR-6-85-5-A, (Universal Robots, Denmark).

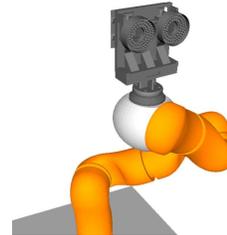


Figure 2. Robot with stereo camera system

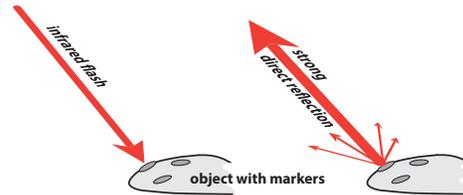


Figure 3. Reflection on object with markers

### 2.2. Image Processing

The circular markers project to ellipses in the image planes. Knowing the calibrated camera parameters, an algorithm detects the centroids of these elliptic shapes with subpixel precision. The centre coordinates of the ellipses are used to calculate a sparse 3D point cloud that gives a discrete representation of the exterior orientation of the tracked object. In order to perform a reliable high-accuracy object tracking, the object feature detection needs to be robust and live. This is done by processing the rectified images from both cameras in parallel. An online processing pipeline individually searches for the ellipses by detecting edges in both images separately with the well-studied edge detector from [5] in a differential formulation for the non-maximum suppression as proposed by [14].

#### 2.2.1 Ellipse Fitting

After noise rejection and the specification of regions of interest (ROIs), an ellipse extraction algorithm finds their cen-

tres. This task is mathematically formulated as a generalized eigenvalue problem similar to the approach of [17].

We start with the representation of a general projectively transformed circle as a quadratic form in  $\mathbb{RP}^2$ . Following the idea of [22], we intersect this conic with the line at infinity to classify object properties. Dehomogenized this reads as

$$\begin{aligned} 0 &= \mathbf{p}^T \mathbf{C} \mathbf{p} \\ &= (a \ b \ c \ d \ e \ f) (x^2 \ xy \ y^2 \ x \ y \ 1)^T \\ &= \mathbf{r}^T \mathbf{d} =: C(\mathbf{r}, \mathbf{d}), \end{aligned} \quad (1)$$

where  $\mathbf{r}$  represents the vector of conic parameters that determine the conic type and  $\mathbf{d}$  is the designed variable vector that describes the structure of the object. The ellipse constraint is given by

$$b^2 - 4ac < 0. \quad (2)$$

$C(\mathbf{r}, \mathbf{d})$  gives the algebraic distance of a point  $p = (x, y, 1)$  to the conic  $C(\mathbf{r}, \mathbf{d}) = 0$ , the basis for our ellipse fitting.

We note that the set of parameters is a homogeneous quantity, since

$$C(\mathbf{r}, \mathbf{d}) = 0 \Leftrightarrow C(\tau \mathbf{r}, \mathbf{d}) = 0 \quad \forall \tau \in \mathbb{R} \setminus \{0\}. \quad (3)$$

This leaves the choice of arbitrarily scaling the parameter vector  $\mathbf{r}$ . We follow the idea of [8] and incorporate the choice of  $\tau$  into the inequality ellipse constraint. We rewrite this in terms of matrix-vector formalisms as

$$\begin{aligned} 1 &= 4ac - b^2 = \mathbf{r}^T \mathbf{A} \mathbf{r} \\ &= (a \ b \ c \ d \ e \ f) \underbrace{\begin{pmatrix} 0 & 0 & 2 & & & \\ 0 & -1 & 0 & \mathbf{0}_{[3,3]} & & \\ 2 & 0 & 0 & & & \\ & \mathbf{0}_{[3,3]} & & \mathbf{0}_{[3,3]} & & \end{pmatrix}}_{\mathbf{A}} \begin{pmatrix} a \\ b \\ c \\ d \\ e \\ f \end{pmatrix} \end{aligned} \quad (4)$$

with  $\mathbf{0}_{[3,3]}$  representing a  $3 \times 3$  matrix filled with zeros.

Let us now write all measurements in one matrix. If  $n$  measured points are given by  $p_i = (x_i, y_i, 1)$ , we write them in form of a vector  $\mathbf{d}$  and construct

$$\mathbf{D} = \begin{pmatrix} \mathbf{d}_1^T \\ \vdots \\ \mathbf{d}_n^T \end{pmatrix} = \begin{pmatrix} x_1^2 & x_1 y_1 & y_1^2 & x_1 & y_1 & 1 \\ & & \vdots & & & \\ x_n^2 & x_n y_n & y_n^2 & x_n & y_n & 1 \end{pmatrix}. \quad (5)$$

Then the algebraic minimization problem summing up all the residuals reduces to

$$\min \|\mathbf{D} \mathbf{r}\|^2 \quad \text{subject to} \quad \mathbf{r}^T \mathbf{A} \mathbf{r} = 1. \quad (6)$$

Solving this with a Lagrange multiplier leads to the necessary condition

$$\mathbf{0} \stackrel{\dagger}{=} \nabla \left( \|\mathbf{D} \mathbf{r}\|^2 - \lambda (\mathbf{r}^T \mathbf{A} \mathbf{r} - 1) \right) = 2 \mathbf{D}^T \mathbf{D} \mathbf{r} - 2\lambda \mathbf{A} \mathbf{r}. \quad (7)$$

This equation can be written as the generalized eigenvalue problem

$$\mathbf{M} \mathbf{r} = \lambda \mathbf{A} \mathbf{r}, \quad \text{where} \quad \mathbf{M} := \mathbf{D}^T \mathbf{D}. \quad (8)$$

There are six eigenvalue-eigenvector pairs  $(\lambda_k, \mathbf{r}_k)$  and since

$$\|\mathbf{D} \mathbf{r}\|^2 = \mathbf{r}^T \mathbf{D}^T \mathbf{D} \mathbf{r} = \mathbf{r}^T \mathbf{M} \mathbf{r} = \lambda \mathbf{r}^T \mathbf{A} \mathbf{r} = \lambda, \quad (9)$$

we are interested in the eigenvector  $\mathbf{r}_+$  which corresponds to the minimal eigenvalue  $\lambda_+ \in \mathbb{R}_0^+$  for which the minimization achieves the best value. Moreover, such an eigenvalue is unique and it always exists.<sup>1</sup> Calculation of this eigenvector allows to robustly fit the ellipses to our data even in case of partial marker occlusion such that we are able to triangulate the 3D marker centres and thus generate a point cloud.

## 2.2.2 Stereo Matching

Having calculated the ellipses, we take their centres to calculate a sparse 3D point cloud. Therefore the stereo correspondences of the subpixel accurate centres are needed. We restrict the search space to the epipolar line including a small neighbourhood and denote corresponding point pairs by  $\mathbf{x}_L \leftrightarrow \mathbf{x}_R$ . For the stereo matching, we follow some ideas of [18] and introduce another constraint on the point set: the boundedness of the disparity gradient.

Let us investigate two points  $\mathbf{x}_W^1, \mathbf{x}_W^2$  given in 3D which project to  $\mathbf{x}_L^i = (x_{Lx}^i, x_y^i)$  in the left image  $\mathbf{I}_L$  and to  $\mathbf{x}_R^i = (x_{Rx}^i, x_y^i)$  in the right rectified image  $\mathbf{I}_R$ ,  $i \in \{1, 2\}$ . We can define a cyclopean image  $\mathbf{I}_C$  by averaging the coordinate values, which gives the coordinates

$$\mathbf{x}_C^i = \left( \frac{x_{Lx}^i + x_{Rx}^i}{2}, x_y^i \right), \quad i \in \{1, 2\}. \quad (10)$$

These image coordinates are shown in fig.4.

The distance of the points in the cyclopean image is called the cyclopean separation  $S(\mathbf{x}_W^1, \mathbf{x}_W^2)$  of  $\mathbf{x}_W^1$  and  $\mathbf{x}_W^2$ . Together with the disparity difference  $D(\mathbf{x}_W^1, \mathbf{x}_W^2)$  we can formulate the disparity gradient  $\Gamma(\mathbf{x}_W^1, \mathbf{x}_W^2)$  as their ratio

$$\Gamma(\mathbf{x}_W^1, \mathbf{x}_W^2) = \frac{D(\mathbf{x}_W^1, \mathbf{x}_W^2)}{S(\mathbf{x}_W^1, \mathbf{x}_W^2)}. \quad (11)$$

<sup>1</sup>It can be shown that the signs of the eigenvalues of the generalized eigenvalue problem  $\mathbf{M} \mathbf{r} = \lambda \mathbf{A} \mathbf{r}$  with positive definite  $\mathbf{M}$  and symmetric  $\mathbf{A}$  are the same as the signs of the eigenvalues of  $\mathbf{A}$  up to permutation (see Lemma 1, [8]). Since the different eigenvalues of  $\mathbf{A}$  in our case are  $\{-2, -1, 0, 2\}$ , this leaves exactly one positive eigenvalue.

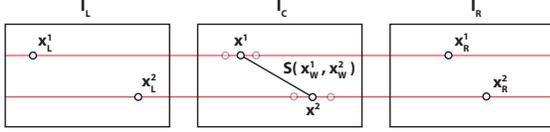


Figure 4. Cyclopean image of  $\mathbf{x}_W^1$  and  $\mathbf{x}_W^2$  with cyclopean separation  $S(\mathbf{x}_W^1, \mathbf{x}_W^2)$

As mentioned by [28], the disparity gradient can be expected to be limited with  $\Gamma(\mathbf{x}_W^1, \mathbf{x}_W^2) \in [-1, 1]$ . We use this to score all possible matches according to all other point pairs that either support the pairing if  $|\Gamma(\mathbf{x}_W^1, \mathbf{x}_W^2)| \leq 1$  or not. Weighting this score by the reciprocal distance from the considered match, we extract world coordinates from rectified stereo images by always taking the matched pair with the highest score.

### 2.3. Tracking

We use these 3D points to estimate the 6 degrees of freedom (DOFs) for an arbitrary motion of the forearm. We track the movement by registering the measured sparse point clouds to a learned point cloud describing the object. Common dense point cloud techniques such as variants of the Iterative Closest Point algorithms as discussed by [23] are not favourable due to the lack of an initial estimation and the number of points. We thus model the pose estimation formally as an energy functional minimization and explain an algorithm to solve the optimisation in real-time.

#### 2.3.1 Energy functional

We have two point clouds to match in every frame pair such that transforming the first with the estimated pose approximates the second. Due to occlusions or measurement errors, the clouds are not necessarily of the same size. Thus a one-to-one correspondence is not always guaranteed. We call these clouds

$$X = \{\mathbf{x}_j \in \mathbb{R}^3 \mid 1 \leq j \leq J\}, \quad (12)$$

$$Y = \{\mathbf{y}_k \in \mathbb{R}^3 \mid 1 \leq k \leq K\}. \quad (13)$$

Some example correspondences  $\mathbf{x}_j \leftrightarrow \mathbf{y}_k$  are shown in fig.5 as connecting lines between the points of the two sets. Not every point has a partner. An estimated transformation maps the blue points of set  $X$  onto the green points which lie most likely close to the red points of set  $Y$ .

Considering these point sets we can now look at the point correspondences and find out which relative pose transforms one cloud into the other. Both tasks are connected in the sense that the answer to one facilitates the other.

The transformation can be modelled by a pose consisting of a rotation matrix  $\mathbf{R} \in \mathbb{R}^{3 \times 3}$  and a translation  $\mathbf{t} \in \mathbb{R}^3$ . The correspondence can be expressed by a permutation matrix

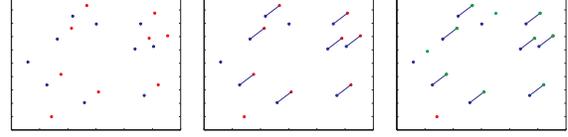


Figure 5. Two point clouds with correspondence and pose estimation

for which also 0-rows and -columns are allowed. We define this match-matrix  $\mathbf{M}$  with entries  $m_{jk}$  by

$$m_{jk} = \begin{cases} 1, & \text{if } \mathbf{x}_j \leftrightarrow \mathbf{y}_k \\ 0, & \text{otherwise.} \end{cases} \quad (14)$$

This leads to a minimization problem of an energy functional ( $\min E(\mathbf{M}, \mathbf{R}, \mathbf{t})$ ) as described by [10] where the energy  $E$  can be expressed as

$$E(\mathbf{M}, \mathbf{R}, \mathbf{t}) = \sum_{j=1}^J \sum_{k=1}^K m_{jk} \|\mathbf{y}_k - (\mathbf{R}\mathbf{x}_j + \mathbf{t})\|^2 - \alpha \sum_{j=1}^J \sum_{k=1}^K m_{jk}. \quad (15)$$

The first term with the Euclidean norm  $\|\cdot\| = \|\cdot\|_2$  gives the distance of the estimation  $\mathbf{R}\mathbf{x}_j + \mathbf{t}$  from the corresponding point  $\mathbf{y}_k$  which we want to minimize. Since the trivial solution of this alone would be a non-correspondence scenario, we also use a second term which pushes the system towards matches. The parameter  $\alpha$  can be understood as a control parameter for the noise toleration of the equation.

The constraints of this minimization problem arise from the definition of the match-matrix and are in particular

$$\sum_k m_{jk} \leq 1 \quad \forall j \in \{1, 2, \dots, J\}, \quad (16)$$

$$\sum_j m_{jk} \leq 1 \quad \forall k \in \{1, 2, \dots, K\} \quad (17)$$

and  $m_{jk} \in \{0, 1\}$ . The first inequality guarantees that every point  $\mathbf{x}_j$  has at most one corresponding partner in the set  $Y$ , while the second one makes sure that every point  $\mathbf{y}_k$  has not more than one partner in  $X$ . At last, the binary constraint assures that there is either a correspondence or not. This gives a mixed minimization problem with a continuous part in the energy functional and a discrete part within the constraints. Furthermore, the constraints consist of two inequalities. In total we have 6 DOFs for the pose and  $J \cdot K$  decisions for the entries of the match matrix.

General solvers exist for problems like this, though real-time processing with such a programme is not possible if it does not take the special framework of the scenario into account. In order to develop a method that is able to solve the minimization in real time, we transform several pieces of the original problem.

### 2.3.2 Constraint Adjustment

First, we convert the inequalities into equalities by reshaping of  $\mathbf{M}$ . Then, the whole problem is translated to a fully continuous setting with a discrete counterpart.

We begin by appending slack variables as proposed by [11] to row  $\mathbf{s}_r$  and column  $\mathbf{s}_c$  of the matrix  $\mathbf{M}$ . This gives

$$\hat{\mathbf{M}} = \left( \begin{array}{c|c} \mathbf{M} & \mathbf{s}_c \\ \hline -\mathbf{s}_r & \end{array} \right) \quad (18)$$

and we can replace the inequalities (16) and (17) in the constraints with equalities to represent a normalization constraint on the rows and columns of the matrix. The slack is 0 except for the case of no corresponding points in the other set. In this case, the additional entry becomes 1.

Furthermore, interpreting the entries  $\hat{m}_{jk}$  of the matrix  $\hat{\mathbf{M}}$  as probabilities for a correspondence or non-correspondence, our match-matrix becomes a doubly stochastic matrix  $\bar{\mathbf{M}}$  as discussed by [25], where

$$\bar{m}_{jk} \in [0, 1] \quad \forall j \in \{1, \dots, J+1\}, k \in \{1, \dots, K+1\}. \quad (19)$$

This gives a possibility to describe the entire problem in a continuous manner. The discrete case is then a special case of this, where the probability of two points being partners is either 1 or 0.

### 2.3.3 Mutual Approximation Updates

In the end, we are not interested in the correspondence probabilities of all different point pairs, but rather want to fit the discrete object representation to the current measurement cloud. Fig.6 illustrates the general algorithmic pipeline of this approach which restarts with every stereo acquisition.

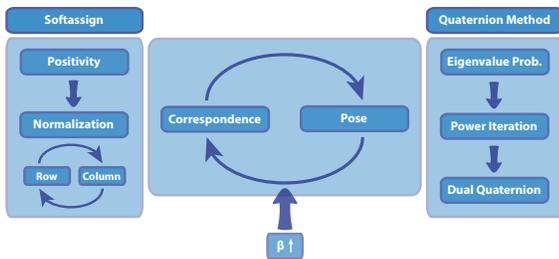


Figure 6. Overview of tracking algorithm

We update the approximation for the stochastic correspondence matrix  $\bar{\mathbf{M}}$  with all the information of the pose we already have, which might be none in the beginning. With this approximation we then calculate a new estimation for the pose which gives the input for the next iteration.

Since the estimation becomes more precise with increasing iterations, an early approximation is ranked with a low confidence  $\beta$  that increases with the number of iterations. Formally speaking, we want the convergence

$$\bar{\mathbf{M}}(\beta) \xrightarrow{\beta \rightarrow \infty} \hat{\mathbf{M}} \quad (20)$$

with the control parameter  $\beta$  that represents the confidence level. This parameter is reinitialized with  $\beta_0$  for a newly acquired frame pair and increments with  $\beta = \beta\beta_{inc}$  in every loop until  $\beta_{max}$  is reached.

As a next step we model both parts, the correspondence and the pose estimation within such an iteration loop separately following the left and right parts of the illustration in fig.6.

### 2.3.4 Correspondence Estimation

A softmax process as described by [3] models the correspondence approximation. With the distance error

$$d_{jk} := \|\mathbf{y}_k - (\mathbf{R}\mathbf{x}_j + \mathbf{t})\|^2, \quad (21)$$

differentiation of the energy functional with respect to  $m_{jk}$  gives

$$Q_{jk} := \frac{\partial E}{\partial m_{jk}} = d_{jk}^2 - \alpha. \quad (22)$$

The small values of  $Q_{jk} \in [-\alpha, 0]$  for point pairs are tolerated. Outside of this, the value  $Q_{jk} > 0$  increases as the error distance enlarges. As a start, we assign strictly positive values to the represented point combinations according to the rank ordering of  $Q_{jk}$ . In addition, we scale the negative value of  $Q_{jk}$  with the parameter  $\beta > 0$  which yields

$$q_{jk} := \exp(-\beta Q_{jk}) = \exp(-\beta(d_{jk}^2 - \alpha)). \quad (23)$$

The value of  $q_{jk}$  is small for non-corresponding points and big for corresponding pairs. If we normalize this by the sum of the row entries, for example, we get

$$\frac{\exp(-\beta Q_{jk})}{\sum_j \exp(-\beta Q_{jk})}, \quad (24)$$

which takes value 1 for the maximal value of the row as  $\beta \rightarrow \infty$ . All other values become 0 as  $\beta \rightarrow \infty$ . This method can be seen as the iterative counterpart for a maximization along the row by increasing the parameter  $\beta$ . However, a matrix normalization is not only desired across all rows, but at the same time across all columns. Fortunately, the entries of the matrix consisting of  $q_{jk}$  are all strictly positive and we can thus iteratively normalize the rows and the columns alternately to get a stochastic matrix

with slacks that satisfies the normalization constraints.<sup>2</sup> The use of softmax with alternate normalization is often called softassign [12] and is done for every update of the correspondence shown in fig.6.

Let us now focus on the other side of the approximation: the pose estimation.

### 2.3.5 Pose Estimation

This time, we imagine that  $E$  only depends on the six parameters given by  $\mathbf{R}$  and  $\mathbf{t}$  since the current correspondence estimation from section 2.3.4 fixes the entries of  $\mathbf{M}$  for the moment. This turns the energy functional into

$$E(\mathbf{R}, \mathbf{t}) = \sum_{j,k} m_{jk} \|\mathbf{y}_k - (\mathbf{R}\mathbf{x}_j + \mathbf{t})\|^2 + C \quad (25)$$

with the constant  $C \in \mathbb{R}$ . Since we are only interested in minimizing this functional, we neglect the scalar  $C$  which does not change the solution.

We use quaternions to formulate the minimization of the energy functional as an eigenvalue problem. Similar to the representation of rotations by quaternions of unit length we use dual quaternions of unit length to represent rigid motions in space.

Let us write the dual quaternion  $\hat{\mathbf{q}}$  as an ordered set of quaternions with dual numbers as coefficients such that

$$\hat{\mathbf{q}} = \mathbf{r} + \mathbf{s}\varepsilon = \mathbf{r} + \varepsilon\mathbf{s}, \quad (26)$$

where  $\mathbf{r}, \mathbf{s} \in \mathbb{H}$  for the quaternion algebra  $\mathbb{H}$  and  $\varepsilon^2 = 0$ . The eight parameters for the quaternion entries  $\mathbf{r}$  and  $\mathbf{s}$  reduce to the six DOFs of a pose if we constraint  $\hat{\mathbf{q}}$  to unit length since

$$1 = \hat{\mathbf{q}}^2 = \mathbf{r}^T\mathbf{r} + \varepsilon(\mathbf{r}^T\mathbf{s}). \quad (27)$$

An elegant way to rewrite a pose in terms of dual unit quaternions is given by [29]. We can use the two matrices that arise from the quaternion  $\mathbf{r} \in \mathbb{H}$

$$\mathbf{P}(\mathbf{r}) = \begin{pmatrix} r_4 & -r_3 & r_2 & r_1 \\ r_3 & r_4 & -r_1 & r_2 \\ -r_2 & r_1 & r_4 & r_3 \\ -r_1 & -r_2 & -r_3 & r_4 \end{pmatrix} \quad \text{and} \quad (28)$$

$$\mathbf{W}(\mathbf{r}) = \begin{pmatrix} r_4 & r_3 & -r_2 & r_1 \\ -r_3 & r_4 & r_1 & r_2 \\ r_2 & -r_1 & r_4 & r_3 \\ -r_1 & -r_2 & -r_3 & r_4 \end{pmatrix} \quad (29)$$

<sup>2</sup>This is due to theorem 2 by [25] which guarantees the convergence to such a stochastic matrix. It is only formulated for square matrices. To use this result, we can embed our problem within a larger problem by extending the smaller dimension of the stochastic matrix. Choosing all suchlike created entries to be zero except for the slacks then makes the theorem applicable in our case and we can forget about the added entries afterwards.

to reformulate the rigid transformations

$$\begin{pmatrix} \mathbf{R} & \mathbf{0} \\ \mathbf{0}^T & 1 \end{pmatrix} = \mathbf{W}(\mathbf{r})^T \mathbf{P}(\mathbf{r}) \quad \text{and} \quad \begin{pmatrix} \mathbf{t} \\ 0 \end{pmatrix} = \mathbf{W}(\mathbf{r})^T \mathbf{s}. \quad (30)$$

Using point quaternions  $\mathbf{p} = (\mathbf{p} \ 0)^T$  for points  $\mathbf{p} \in \mathbb{R}^3$  and division by 2 gives the reformulated energy functional

$$\begin{aligned} E(\mathbf{r}, \mathbf{s}) &= \mathbf{r}^T \left( -\sum_{j,k} m_{jk} \mathbf{P}(\mathbf{y}_k)^T \mathbf{W}(\mathbf{x}_j) \right) \mathbf{r} \\ &+ \mathbf{s}^T \left( \frac{1}{2} \sum_{j,k} m_{jk} \mathbf{I} \right) \mathbf{s} \\ &+ \mathbf{s}^T \left( \sum_{j,k} m_{jk} (\mathbf{W}(\mathbf{x}_j) - \mathbf{P}(\mathbf{y}_k)) \right) \mathbf{r} \\ &=: \mathbf{r}^T \mathbf{C}_1 \mathbf{r} + \mathbf{s}^T \mathbf{C}_2 \mathbf{s} + \mathbf{s}^T \mathbf{C}_3 \mathbf{r} \end{aligned} \quad (31)$$

where constant terms have been neglected since they do not change the minimization. Incorporating the unit constraints as Lagrange multipliers we get

$$E(\mathbf{r}, \mathbf{s}) = \mathbf{r}^T \mathbf{C}_1 \mathbf{r} + \mathbf{s}^T \mathbf{C}_2 \mathbf{s} + \mathbf{s}^T \mathbf{C}_3 \mathbf{r} + \lambda_1 (\mathbf{r}^T \mathbf{r} - 1) + \lambda_2 (\mathbf{r}^T \mathbf{s}) \quad (32)$$

and the necessary condition for the minimization reads as

$$\nabla E = (\partial_{\mathbf{r}} E, \partial_{\mathbf{s}} E)^T \stackrel{!}{=} \mathbf{0}. \quad (33)$$

Calculating the partial derivatives reveals  $\lambda_2 = 0$  and we conclude with the eigenvalue problem

$$\mathbf{A}\mathbf{r} = \lambda_1 \mathbf{r} \quad \text{where} \quad \mathbf{A} = \frac{1}{4} \mathbf{C}_3^T \mathbf{C}_2^{-1} \mathbf{C}_3 - \mathbf{C}_1. \quad (34)$$

Replacing the terms for  $\mathbf{C}_1$  and  $\mathbf{C}_2$  in the energy functional finally yields

$$E(\mathbf{r}, \mathbf{s}) = -\lambda_1 \quad (35)$$

which is minimal for the maximal eigenvalue  $\lambda_1$  of  $\mathbf{A}$ . To find the dominant eigenvector corresponding to the largest eigenvalue, we do not have to calculate all eigenvalues separately. Since the matrix  $\mathbf{A}$  is real and symmetric, the eigenvectors are orthogonal. For a speed-up of this calculation, we can therefore use a power iteration. For a non-degenerated start quaternion  $\bar{\mathbf{r}}_0$ , the sequence

$$\bar{\mathbf{r}}_{n+1} = \frac{\mathbf{A}\bar{\mathbf{r}}_n}{\|\mathbf{A}\bar{\mathbf{r}}_n\|} \xrightarrow{n \rightarrow \infty} \mathbf{r}_{max} \quad (36)$$

converges to this normalized dominant eigenvector. Finally we can determine the dual quaternion part  $\mathbf{s}_{max}$  from  $\mathbf{r}_{max}$  and resubstitution gives the rotation  $\mathbf{R}$  and a translation vector  $\mathbf{t}$  for each update of the pose in fig.6.

### 2.3.6 Object Teaching

A fast one to multi-shot learning procedure of a new marker set-up ensures consistently precise measuring results with a generic marker ensemble. In teaching mode, the object point cloud is transformed to a position-independent reference coordinate system given at the start of the learning phase by a translation of the camera coordinate system to the centroid of the initial observation. All consecutive measurements are registered to this initial cloud and are transformed to the same coordinate system while adjusting the model parameters and rejecting outliers on the fly.

This enables the user to explicitly decide on the training duration while movement and partial marker occlusion are not problematic in the teaching phase.

### 2.4. Robot Control

The robot control is based on the Fast Research Interface (FRI) provided by KUKA, Germany and extended by the ROS framework [20] to support path planning. It leverages the TF library that allows for an accurate and real-time knowledge of the positions of the robot and the healthy arm, which is continuously updated with tracking information. To this end, the OpenIGTLink API [26] ensures the TCP/IP-based communication between the computers hosting the robot-control and tracking applications. A virtual reference frame attached to the marker represents the target position for the end-effector of the robot. This transformation is found by manually bringing the system to the desired position relative to the healthy arm, and then querying the current pose.

The control loop keeps planning the trajectory required to bring the end effector of the robot to the null position in the virtual target reference frame; this has the effect of moving the robot to the desired position relative to the healthy arm. MoveIt! and OMPL are employed for the path planning and self-collision avoidance; the planning algorithm used is a variant of randomly exploring random trees, RRT-Connect. In order to avoid undesired trajectories that could arise given the stochastic nature of the algorithm and for safety reasons while moving in close proximity to the user, we check the movement range of each joint and reject trajectories which make them move beyond a threshold. The resulting joint trajectory is asynchronously pushed to an instance of the joint-trajectory-action controller from ROS, which forwards it to the FRI. The trajectory is updated every 40 ms. The controller performs a quintic spline interpolation with the current state, ensuring acceleration-level continuity. This control architecture enables a fluid and natural movement of the robot.

For safety, the velocity and acceleration of the robot are limited to 0.24 mm/s and 0.8 mm/s<sup>2</sup>.

## 3. Experiments and Results

### 3.1. Tracking Parameters

The parameters for the tracking algorithm as described in section 2.3.3 are determined empirically. We use  $\beta_0 = 10^{-4}$ ,  $\beta_{inc} = 1.053$ , and  $\beta_{max} = 10^3$ . In our tests, we always noticed convergence within the first 5 loops and thus set the iteration maximum for the mutual updates to  $I_{max} = 10$  and accept fits as long as 50 % of all markers have corresponding partners and the root mean square error of the transformation is smaller than 3 mm. The teaching is done once initially and repeated online if the marker deformation on the test person's skin causes these constraints to reject multiple measurements in a row.

### 3.2. Accuracy, Robustness and Latency

The first two experiments analyse the accuracy and robustness of the tracking system and the robot movement. The optical system is calibrated for its intrinsic and exterior parameters. The overall mean reprojection error for this calibration is 0.29 pixels.

The repeatability of the robot pose, according to the manufacturer, is  $\pm 0.05$  mm [13]. It serves as ground truth to quantify the error of the whole tracking system (i.e. camera calibration, pose estimation and hand-eye calibration), measured by moving the robot in the relevant working volume - a cuboid of  $60 \times 60 \times 40$  mm<sup>3</sup> - along a planned trajectory in steps of 20 mm while simultaneously observing a target on a table parallel to the xy-plane in robot base coordinates. The root mean square error for the pose fitting is 0.21 mm  $\pm 0.25$  mm given from the estimates in camera coordinates. The standard deviation for the total translation error of the observed object reference frame is measured as 0.23 mm, 0.23 mm, 0.42 mm in x, y, and z. To further investigate the robustness of the system, up to 50 % of the markers are covered in the same experiment while the system is still able to register the point clouds correctly.

The processing time for the pose estimation with 10 markers on the forearm is 9.42 ms  $\pm 1.44$  ms. The overall system latency is tested with a second robot that holds a target observed by the first robot. Both robots are co-calibrated and the first follows the second while the trajectories are recorded. Fig.7 shows the translational component of the robots during one move. We calculate the delay between both movements as the time difference between the beginning of each robot's movements, indicated by the dashed vertical lines. The latency is 318.70 ms, which includes all system components.

### 3.3. Movement Therapy

Two healthy medical experts evaluate the usability of the device. As shown in the video provided as supplementary

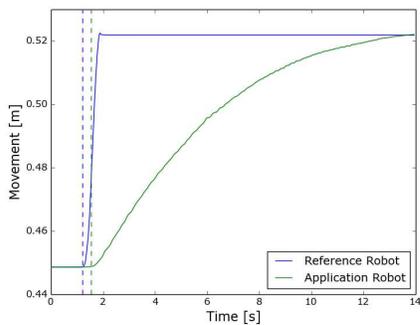


Figure 7. Robot movement latency tests

material<sup>3</sup>, they rest their left arm in the sling and are asked to suppress any voluntary movement of this arm. They execute a series of movements with their right arm in all three directions and are asked to provide feedback on their experience. The feedback from both testers is that the robot repeats the movement of the right with the left, fully relaxed arm in the sleeve. Steering the robot with natural arm movements is described as being a very intuitive control mechanism.

#### 4. Discussion

We introduce a camera-in-hand based tracking solution on a light-weight robot to observe the movement of the healthy hand and reproduce it with the impaired hand. While the concept of collaborative robotics has been introduced for rehabilitation tasks with joystick- and exoskeleton robots [1], our system allows controlling the robot with natural gestures. It does neither constrain the healthy hand nor requires manipulating an unnatural object to inform the robot about the movement to be carried out. Touchless human-computer and, in its extension, human-robot interaction is a vast field of research, however the main challenges lie in providing a robust and real-time recognition of the human intent [21]. The method we present uses a marker-based algorithm whose main disadvantage is the necessity of using markers. These markers however ensure robust, accurate and fast tracking with a total system latency of 318.70 ms and a precision better than 0.5 mm in the relevant area necessary to provide a natural experience with the robot which are not achieved by markerless methods [24], [4]. This performance compares with the state of the art in marker-based tracking [9]. However, it is more flexible to use since the overall marker structure is not restricted to one plane as for many common fiducial marker systems [16], [30] while it provides much faster tracking information, [7]. The fast learning algorithm allows the system to learn new marker configurations on

<sup>3</sup>Video available for download at <http://campar.in.tum.de/Chair/PublicationDetail?pub=busam2015acvr>.

the fly, allowing the simple exchange of the sleeve and relearning the marker configuration even if the sleeve moves during a training session.

Being a first concept demonstrator, the presented system is able to repeat only very simple movements as compared to other upper limb rehabilitation robots [1]. It allows only for the combined training of the forearm and shoulder movements in 3 directions, without any ability to decompose the movement or explicitly model elbow bending. In the current state, it also does not allow adapting the force applied to the impaired arm and the robot movement speed is limited for safety reasons. Since the tracking algorithm is able to simultaneously track several targets in real-time and the cameras have a wide field of view, it is possible to extend this approach to several targets mounted on different parts of the arm to track their exact movement. The manipulator needs redesign towards a rigidly attached exoskeleton to reproduce all possible arm movements.

The use of a camera-in-hand tracking system reduces the problems of interrupting the line of sight between the camera and the markers that would be present using external cameras. The placement of two cameras on the robotic arm gives it a more humanoid appearance, likely to contribute to the robots acceptance by patients.

Tests with healthy individuals confirm the feasibility of this approach. In order to assess its advantages and clinical impact, above-mentioned developments towards full rehabilitation system are required before entering a clinical trial with impaired patients. Finally, the influence of this system on the patient outcome can be compared to other robotic and non-robotic rehabilitation approaches.

#### 5. Conclusion

This work presents a collaborative robotic system for upper limb movement therapy in hemiparetic patients. It introduces a robust real-time marker-based tracking algorithm that provides the necessary information to reproduce the movement of the healthy arm with the impaired one. While the accuracy of the tracking concept is confirmed, first trials with healthy subjects show the feasibility of this approach. Yet, further developments are necessary for a system that trains natural hand movements. Finally, this work opens the way towards gesture-controlled robots in rehabilitation.

#### References

- [1] A. Basteris, S. M. Nijenhuis, A. Stienen, J. H. Burke, G. B. Prange, and F. Amirabdollahian. Training modalities in robot-mediated upper limb rehabilitation in stroke: a framework for classification based on a systematic review. *Journal of neuroengineering and rehabilitation*, 11(1):111, 2014. 1, 8

- [2] A. A. Blank, J. A. French, A. U. Pehlivan, and M. K. O'Malley. Current trends in robot-assisted upper-limb stroke rehabilitation: promoting patient engagement in therapy. *Current physical medicine and rehabilitation reports*, 2(3):184–195, 2014. 1
- [3] J. S. Bridle. Training stochastic model recognition algorithms as networks can lead to maximum mutual information estimation of parameters. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 2*, pages 211–217, 11 1989. 5
- [4] P. Buehler, M. Everingham, D. Huttenlocher, and A. Zisserman. Upper body detection and tracking in extended signing sequences. *International Journal of Computer Vision*, 95(2):180–197, 2011. 8
- [5] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, Nov 1986. 2
- [6] V. L. Feigin, M. H. Forouzanfar, R. Krishnamurthi, G. A. Mensah, M. Connor, D. A. Bennett, A. E. Moran, R. L. Sacco, L. Anderson, T. Truelsen, et al. Global and regional burden of stroke during 1990–2010: findings from the global burden of disease study 2010. *The Lancet*, 383(9913):245–255, 2014. 1
- [7] M. Fiala. Designing highly reliable fiducial markers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(7):1317–1324, July 2010. 8
- [8] A. Fitzgibbon, M. Pilu, and R. B. Fisher. Direct least square fitting of ellipses. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):476–480, 1999. 3
- [9] S. Garrido-Jurado, R. Muñoz-Salinas, F. Madrid-Cuevas, and M. Marn-Jimnez. Automatic generation and detection of highly reliable fiducial markers under occlusion. *Pattern Recognition*, 47(6):2280 – 2292, 2014. 8
- [10] S. Gold, C. P. Lu, A. Rangarajan, S. Pappu, and E. Mjølness. Fast algorithms for 2d and 3d point matching: Pose estimation and correspondence. Research Report YALEU/DCS/RR-1035, Yale University, Department of Computer Science, New Haven. CT 06520-8285, 05 1994. 4
- [11] S. Gold and A. Rangarajan. A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(4):377–388, 1996. 5
- [12] S. Gold, A. Rangarajan, C.-P. Lu, S. Pappu, and E. Mjølness. New algorithms for 2d and 3d point matching: pose estimation and correspondence. *Pattern Recognition*, 31(8):1019 – 1031, 1998. 6
- [13] KUKA GmbH. Kuka lwr. user-friendly, sensitive and flexible. [http://www.kukaconnect.com/wp-content/uploads/2012/07/KUKA\\_LBR4plus\\_ENLISCH.pdf](http://www.kukaconnect.com/wp-content/uploads/2012/07/KUKA_LBR4plus_ENLISCH.pdf), 2012. 7
- [14] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 465–470, Jun 1996. 2
- [15] É. Marchand, F. Spindler, and F. Chaumette. Visp for visual servoing: a generic software platform with a wide class of robot control skills. *IEEE Robotics & Automation Magazine*, 12(4):40–52, 2005. 2
- [16] L. Naimark and E. Foxlin. Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker. In *International Symposium on Mixed and Augmented Reality. Proceedings*, pages 27–36, 2002. 8
- [17] P. O’Leary and P. Zsombor-Murray. Direct and specific least-square fitting of hyperbolae and ellipses. *Journal of Electronic Imaging*, 13(3):492–503, 2004. 3
- [18] S. B. Pollard, J. E. W. Mayhew, and J. P. Frisby. Pmf: A stereo correspondence algorithm using a disparity gradient limit. *Perception*, 14:449–470, 1985. 3
- [19] J. Pulman and E. Buckley. Assessing the efficacy of different upper limb hemiparesis interventions on improving health-related quality of life in stroke patients: a systematic review. *Topics in stroke rehabilitation*, 20(2):171–188, 2013. 1
- [20] M. Quigley, K. Conley, B. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng. ROS: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5, 2009. 7
- [21] S. S. Rautaray and A. Agrawal. Vision based hand gesture recognition for human computer interaction: a survey. *Artificial Intelligence Review*, 43(1):1–54, 2015. 8
- [22] J. Richter-Gebert. *Perspectives on Projective Geometry - A Guided Tour Through Real and Complex Geometry*, volume XXII. Springer, 2011. 3
- [23] S. Rusinkiewicz and M. Levoy. Efficient variants of the icp algorithm. In *Third International Conference on 3-D Digital Imaging and Modeling, 2001. Proceedings*, pages 145–152. IEEE, 2001. 4
- [24] M. Siddiqui and G. Medioni. Robust real-time upper body limb detection and tracking. In *ACM International Workshop on Video Surveillance & Sensor Networks (VSSN)*, 2006. 8
- [25] R. Sinkhorn. A relationship between arbitrary positive matrices and doubly stochastic matrices. *The Annals of Mathematical Statistics*, 35(2):pp. 876–879, 06 1964. 5, 6
- [26] J. Tokuda, G. Fischer, X. Papademetris, Z. Yaniv, L. Ibanez, P. Cheng, H. Liu, J. Blevins, J. Arata, A. Golby, T. Kapur, S. Pieper, E. Burdette, G. Fichtinger, C. Tempany, and N. Hata. Openiglink: An open network protocol for image-guided therapy environment. *The international journal of medical robotics + computer assisted surgery: MRCAS*, 5(4):423–434, Dec 2009. 7
- [27] R. Tsai and R. Lenz. A new technique for fully autonomous and efficient 3D robotics hand/eye calibration. *IEEE Transactions on Robotics and Automation*, 5(3):345–358, June 1989. 2
- [28] M. Šonka, V. Hlavac, and R. Boyle. *Image Processing, Analysis, and Machine Vision*. Thomson Learning, third edition, 2008. 4
- [29] M. W. Walker, L. Shao, and R. A. Volz. Estimating 3-d location parameters using dual number quaternions. *CVGIP: Image Understanding*, 54(3):358 – 367, 1991. 6
- [30] X. Zhang, S. Fronz, and N. Navab. Visual marker detection and decoding in ar systems: a comparative study. In *International Symposium on Mixed and Augmented Reality. Proceedings*, pages 97–106, 2002. 8
- [31] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330–1334, Nov 2000. 2