Strathprints Institutional Repository

Yang, Cheng and Kerr, Andrew and Stankovic, Vladimir and Stankovic, Lina and Rowe, Philip and Cheng, Samuel (2016) Human upper limb motion analysis for post-stroke impairment assessment using video analytics. IEEE ACCESS. pp. 1-10. , http://dx.doi.org/10.1109/ACCESS.2016.2523803

This version is available at http://strathprints.strath.ac.uk/55526/

Strathprints is designed to allow users to access the research output of the University of Strathclyde. Unless otherwise explicitly stated on the manuscript, Copyright © and Moral Rights for the papers on this site are retained by the individual authors and/or other copyright owners. Please check the manuscript for details of any other licences that may have been applied. You may not engage in further distribution of the material for any profitmaking activities or any commercial gain. You may freely distribute both the url (http://strathprints.strath.ac.uk/) and the content of this paper for research or private study, educational, or not-for-profit purposes without prior permission or charge.

Any correspondence concerning this service should be sent to Strathprints administrator: strathprints@strath.ac.uk

Human Upper Limb Motion Analysis for Post-stroke Impairment Assessment using Video Analytics

Cheng Yang, *Student Member, IEEE,* Andrew Kerr, Vladimir Stankovic, *Senior Member, IEEE,* Lina Stankovic, *Senior Member, IEEE,* Philip Rowe, and Samuel Cheng* *Senior Member, IEEE*

***Abstract*—Stroke is a worldwide healthcare problem which often causes long-term motor impairment, handicap, and disability. Optical motion analysis systems are commonly used for impairment assessment due to high accuracy. However, the requirement of equipment-heavy and large laboratory space together with operational expertise, makes these systems impractical for local clinic and home use. We propose an alternative, cost-effective and portable, decision support system for optical motion analysis, using a single camera. The system relies on detecting and tracking markers attached to subject's joints, data analytics for calculating relevant rehabilitation parameters, visualization, and robust classification based on graph-based signal processing. Experimental results show that the proposed decision support system has the potential to offer stroke survivors and clinicians an alternative, affordable, accurate and convenient impairment assessment option suitable for home healthcare and tele-rehabilitation.**

***Index Terms*—Rehabilitation, Graph-based signal processing, Video analytics**

I. Introduction

EMERGING multimedia-based motion analysis systems with optical equipment are being increasingly used for periodical limb impairment assessments during rehabilitation for patients who survived stroke, a world-wide healthcare problem which can cause long-term motor impairment, handicap, and disability to survivors [3], [4]. Autonomous mechanism of these systems with high-fidelity outcome measure is welcomed by clinical practitioners, significantly improving the objectivity and accuracy compared to classical visual observation. For example, laboratory-based optical motion analysis systems [5] with high accuracy and real-time tracking features are available so that interventions such as exercise or ankle foot orthoses can be optimally prescribed. These systems capture motion patterns, namely, joint angles [6], [7], by tracking reflective markers fixed to the skin overlying anatomical landmarks of the subject using multiple infrared cameras, and provide visualization for diagnosis, however, with the sacrifice of the cost, space, and portability, which is thus impractical for local clinics and home use.

In our previous conference paper [8], we propose a cost-effective and portable single-camera motion analysis system for lower-limb (gait) analysis to track three bullseye markers attached to the pelvis and legs. The proposed system shows a significant improvement with respect to a joint color-texture histogram (`JCTH`) approach [9] and a Tracking-Learning-Detection (`TLD`) scheme [10]. The tracking result is then used for $manual$ impairment assessment of stroke survivors via gait analysis.

Motivated by the fact that arm impairment is also a common outcome of stroke [4], [11], building on [8], in this paper, we propose a decision support system for upper limb motion analysis that simultaneously tracks a number of identical bullseye markers, and maps the trajectories of the tracked markers into meaningful information used for rehabilitation assessment. The system comprises a single high-speed camera together with a visualisation module that enables navigating through the captured frames, selecting parameters to present, and comparison with the previous results.

The data analytics part of our solution can be used independently of the capturing module to process autonomously existing reach-to-grasp (RTG) video datasets (see Section III), that contain recordings of RTG movements in the sagittal plane with multiple bullseye markers adhered to the joints of a human body, which are a common alternative to 3D datasets. Note that in 2D video-based clinical kinematic analysis [12], as in the RTG datasets, conventionally black-and-white bullseye markers are used, attached to the skin overlying anatomical landmarks of the subject's pelvis, cervical spine, shoulder, elbow, and wrist (see Fig. 1(a)).

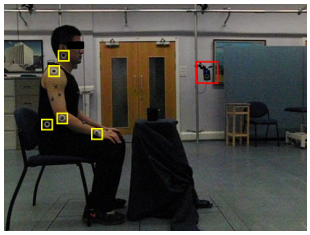
The motion of the subject's upper limb kinematics is captured by tracking the markers frame by frame and autonomously computing joint angles (see Fig. 1(b)). Once the joint angles have been extracted in each frame, they are used as classification features to automatically estimate the level of impairment [13]. Data classification using regularization on graphs [14], [15], [16] is proposed in [17], where it is shown that graph-based supervised binary classification shows competitive performance to conventional classifiers, such as Support Vector Machine (`SVM`) [18], [19] and neural networks, and good robustness to noise in the training dataset. The main idea is to first represent the dataset to be classified as a signal indexed by a graph, whose vertices correspond to samples

C. Yang, V. Stankovic, and L. Stankovic are with the Department of Electronic and Electrical Engineering, University of Strathclyde, Glasgow, UK, G1 1XQ, e-mail: {cheng.yang, vladimir.stankovic, lina.stankovic}@strath.ac.uk.

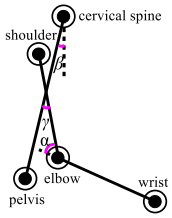
A. Kerr and P. Rowe are with the Department of Biomedical Engineering, University of Strathclyde, Glasgow, UK, G1 1XQ, e-mail: {a.kerr, philip.rowe}@strath.ac.uk.

S. Cheng is with the Department of Computer Science and Technology, Tongji University, Shanghai, China, email: szeming@tongji.edu.cn.

This paper was presented in part at *IEEE ICIP-2014* conference [1] and *WSC-2014* congress [2].



(a) Camera scene



(b) Angles of interest

Fig. 1: Experimental setup.

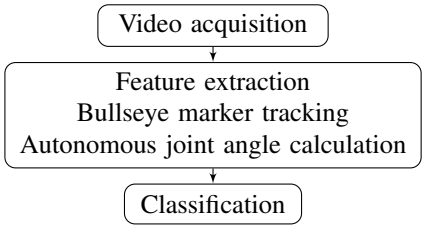


Fig. 2: Unit blocks of the proposed decision support system.

in the dataset and weighted edges reflecting similarities or correlation between vertices, then minimize total variation on a graph [20] based on a binary mapping of this graph. In this paper (see Section II-C), we propose two regularization on graph signals (RGS) based multi-class classification methods, by first constructing graphs for the motion patterns obtained as a result of object tracking, and then designing binary mappings of these graphs using graph-based tools following [17] for minimization of the total variation on graphs [20]. We also propose a third RGS multi-class classification method, by first constructing a graph following [17], and then, designing a multi-class mapping of this graph, unlike binary mappings in [17], [20], and minimize the total variation on graph.

We validate the proposed system with a standardized, multi-infrared-camera Vicon system using a Bland-Altman plot [21], to evaluate the amount of agreement between the two systems. Experimental results show that the proposed system can capture upper limb motion patterns accurately, explicitly classify participants into a healthy group and different stroke groups with levels of impairment [13], provide visual and written feedback, and thus has potential to offer stroke survivors and clinicians an alternative, affordable, accurate and convenient impairment assessment option.

In summary, the main contributions of the paper are:

- Novel multi-class and binary RGS classification methods for rehabilitation diagnostics.
- Effective multimedia-based decision support tools for processing autonomously large RTG video datasets.
- Overall plug-and-play cost-effective motion analysis system suitable for home use, including data capture, processing and visualisation blocks, tested on the patients and designed with the feedback from practitioners.

The remainder of this paper is organized as follows. In the next section we discuss each component of the proposed system. In Section III, we present the experimental results — tracking performance comparison with [9], [10], and [22], angle accuracy validation with state-of-the-art motion analysis system Vicon, and subject classification using RGS. We conclude this paper in Section IV.

II. Proposed System

The aim of the proposed system is to autonomously assess the upper limb motor condition of the subject by accurately and simultaneously tracking the multiple bullseye markers adhered to the joints and provide visual and written feedback to stroke survivors and clinicians.

Impairment of the upper limb following a stroke can be assessed in a number of ways [13], by measuring physical attributes such as range of motion, strength and co-ordination or more commonly by quantitatively assessing the ability to carry out a functional task such as the RTG movement [23], shown in Fig. 1(a), where the subject picks up a cup from the desk, carries it towards the mouth and puts it back on the desk. Three joint angles can be analysed during this activity, namely, (i) elbow movement defined by a supplementary angle to the shoulder-elbow-wrist angle denoted by α shown in Fig. 1(b); (ii) trunk-tilt defined by the pelvis-cervical spine-vertical angle β; and (iii) shoulder movement defined by an angle γ at the intersection of pelvis-cervical spine and shoulder-elbow lines.

To calculate the relevant joint angles, we track, through the captured frames, five bullseye markers adhered to the skin overlying anatomical landmarks of the pelvis, cervical spine, shoulder, elbow, and wrist of the participant, highlighted by yellow squares in Fig. 1(a). The tracked motion patterns are then used to calculate the three angles in each frame, which are subsequently used for classification.

The main components of the proposed human upper limb motion analysis procedure to be described next are shown in Fig. 2.

A. Bullseye marker tracking

Simultaneously tracking all bullseye markers is challenging due to the following marker features: 1) The markers clinically used in 2-D video-based kinematic analysis are *identical* and are in close proximity, which can easily cause tracking confusion. 2) The size, orientation, and appearance of each marker could change due to joints movement, and thus the tracker should be capable of handling such non-rigid objects. 3) These markers move along with the limb motion of the subject, i.e., small-size target objects move with a large moving object that can be assumed as the appearance-changing background [24], which potentially distract marker tracking which reduces the tracking accuracy, and thus the tracker needs to address the *object-on-object* tracking problem.

There is a substantial amount of work on object tracking, and good surveys can be found in [24], [25], [26], [27]. Next, we review the work most relevant to ours. [28] represents each object with object-correspondence-points for point tracking. However, this approach cannot handle non-rigid objects. For silhouette tracking, [29] handles non-rigid objects well by building online shape priors and implementing object contour evolvement using energy minimization in gradient descent direction for target objects. However, [29] is only capable of tracking objects that are very different. [9] embarks from kernel tracking, jointly applies local binary pattern texture with color histogram which effectively extracts the features

of the edges and corners within the target region, and adopts mean-shift with the above JCTH approach and acquires robust performance for tracking objects that have similar color appearance to the background. However, the *object-on-object* problem significantly affects the tracking accuracy and can cause tracking failure. [10] also exploits kernel tracking by online learning and binary classification within a TLD scheme to update the object template adaptively. i.e., [10] is robust for tracking non-rigid objects. However, online learning in [10] is achieved by searching a global frame, which means [10] cannot be directly used for simultaneously tracking multiple objects. "Struck" (STR) [22] is the best tracker among 19 state-of-the-art trackers tested in [27] and a highly competitive online tracker gauged in [25], [30], [31]. The tracking scheme in [22] is based on structured output prediction with kernels. Still, [22] cannot handle out-of-plane rotation well, and there is no object-dynamic model incorporated into this adaptive tracking-by-detection framework. Furthermore, [32] proposes a particle swarm optimization method, and [33] a particle filter-mean shift joint tracking algorithm, both of which achieve simultaneous multiple objects tracking. However, these two methods cannot address the *object-on-object* problem.

In the following, we describe the proposed method that addresses some of the shortcomings of the above approaches for the RTG dataset. First, as in [8], the centre coordinates of all bullseye marker templates are selected via mouse-click on our developed user interface in Frame 1 (see Fig. 3) (the only manual effort during the entire process). All markers are then tracked simultaneously using a Discrete Kalman FIlter (DKF) [34], [35]. First the position and size of a rectangular Search Area (SA) for each marker is set in each frame based on the output of DKF. Then, for each marker, block matching is performed within the SA using structural-similarity (SSIM) [36] to identify a block most similar to the marker template.

Let $\hat{\mathbf{s}}_i^j = [c_i^{\mathrm{SA}_j}, r_i^{\mathrm{SA}_j}, u_i^{\mathrm{SA}_j}, v_i^{\mathrm{SA}_j}]$, where $c_i^{\mathrm{SA}_j}$ and $r_i^{\mathrm{SA}_j}$ denote the column and row of the centre of the SA for Marker j (pelvis, cervical spine, shoulder, elbow, or wrist marker) in Frame i, respectively, and $u_i^{\mathrm{SA}_j}$ and $v_i^{\mathrm{SA}_j}$ are velocities along horizontal and vertical directions, respectively. Similarly, let $\hat{\mathbf{z}}_i^j = [c_i^j, r_i^j]$, where c_i^j and r_i^j denote the column and row of the centre of Marker j in Frame i, respectively.

We use DKF to determine the position and size of SA for all markers in each frame, where $\hat{\mathbf{s}}_i = [\hat{\mathbf{s}}_i^1, \hat{\mathbf{s}}_i^2, \hat{\mathbf{s}}_i^3, \hat{\mathbf{s}}_i^4, \hat{\mathbf{s}}_i^5]^{\mathrm{T}}$ and $\hat{\mathbf{z}}_i = [\hat{\mathbf{z}}_i^1, \hat{\mathbf{z}}_i^2, \hat{\mathbf{z}}_i^3, \hat{\mathbf{z}}_i^4, \hat{\mathbf{z}}_i^5]^{\mathrm{T}}$ are used to build the dynamic and observation model of the DKF, respectively. In particular, the prediction phase is given by: $\hat{\mathbf{s}}_i^- = \mathbf{T}\hat{\mathbf{s}}_{i-1}, \mathbf{P}_i^- = \mathbf{T}\mathbf{P}_{i-1}\mathbf{T}^{\mathrm{T}} + \mathbf{O}$, where $\hat{\mathbf{s}}_i^-$ is the a priori estimate of $\hat{\mathbf{s}}_i$ in Frame i, $\hat{\mathbf{s}}_{i-1}$ is the a posteriori estimate, $\mathbf{T} = diag(\mathbf{T}^1, \mathbf{T}^2, \mathbf{T}^3, \mathbf{T}^4, \mathbf{T}^5)$ is the state transition matrix with $\mathbf{T}^j = [1, 0, t, 0; 0, 1, 0, t; 0, 0, 1, 0; 0, 0, 0, 1]$, where t is the duration of one frame, $\mathbf{P}_i^-$ is the a posteriori covariance matrix, and $\mathbf{O}$ is the process noise covariance matrix pre-computed by running the filter off-line based on the assumption that $\mathbf{O}$ is time invariant [34].

The correction phase is given by: $\mathbf{K}_i = \mathbf{P}_i^-\mathbf{Z}^{\mathrm{T}}/(\mathbf{Z}\mathbf{P}_i^-\mathbf{Z}^{\mathrm{T}} + \mathbf{E}), \hat{\mathbf{s}}_i = \hat{\mathbf{s}}_i^- + \mathbf{K}_i(\hat{\mathbf{z}}_i - \mathbf{Z}\hat{\mathbf{s}}_i^-), \mathbf{P}_i = (\mathbf{I} - \mathbf{K}_i\mathbf{Z})\mathbf{P}_i^-$, where $\mathbf{K}_i$ is the Kalman gain, $\mathbf{Z} = diag(\mathbf{Z}^1, \mathbf{Z}^2, \mathbf{Z}^3, \mathbf{Z}^4, \mathbf{Z}^5)$ is

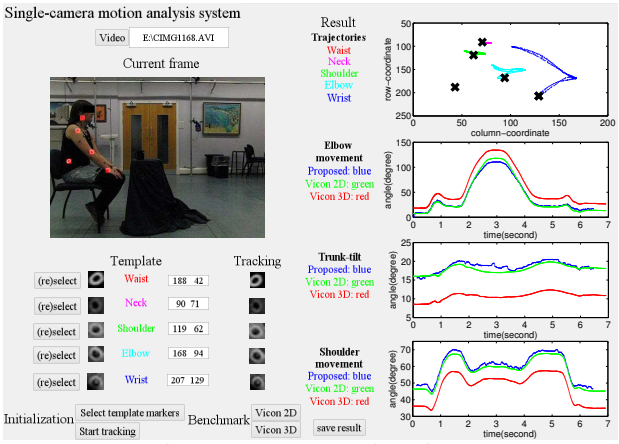


Fig. 3: System user interface.

the observation matrix which translates $\hat{\mathbf{s}}_i$ to $\hat{\mathbf{z}}_i$, with $\mathbf{Z}^j = [1, 0, 0, 0; 0, 1, 0, 0]$, and $\mathbf{E}$ is the measurement error covariance matrix pre-computed by running the filter off-line based on the assumption that $\mathbf{E}$ is constant across all frames [34].

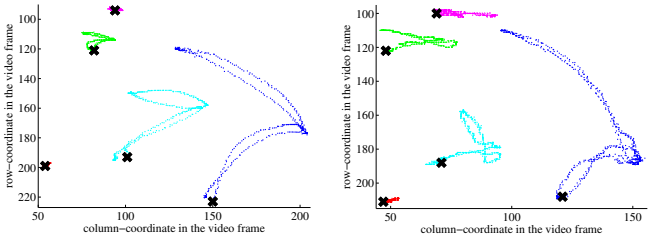
We initialize the DKF by $\hat{\mathbf{s}}_1^- = [\hat{\mathbf{s}}_0^1, \hat{\mathbf{s}}_0^2, \hat{\mathbf{s}}_0^3, \hat{\mathbf{s}}_0^4, \hat{\mathbf{s}}_0^5]^{\mathrm{T}}$, with $\hat{\mathbf{s}}_0^j = [c_0^{\mathrm{SA}_j}, r_0^{\mathrm{SA}_j}, 0, 0]$, and $\hat{\mathbf{z}}_1 = [\hat{\mathbf{z}}_1^1, \hat{\mathbf{z}}_1^2, \hat{\mathbf{z}}_1^3, \hat{\mathbf{z}}_1^4, \hat{\mathbf{z}}_1^5]^{\mathrm{T}}$, with $\hat{\mathbf{z}}_1^j = [c_1^j, r_1^j]$, where (c_1^j, r_1^j) is the centre coordinate of the marker template, and $c_0^{\mathrm{SA}_j} = c_1^j$, $r_0^{\mathrm{SA}_j} = r_1^j$.

The size of each SA is initialized to $\lfloor 1.4q \rfloor \times \lfloor 1.4q \rfloor$ pixels, given Marker j's size is $q \times q$ pixels, and is dynamically updated, for each marker, in each frame according to $u_i^{\mathrm{SA}_j}$ and $v_i^{\mathrm{SA}_j}$ firstly by adjusting the width (horizontally) and then updating the height (vertically). If $u_i^{\mathrm{SA}_j} t \geq 0$ the width of the SA is increased from $\lfloor 1.4q \rfloor$ pixels to $\lfloor 1.4q \rfloor + u_i^{\mathrm{SA}_j} t$ horizontally towards the right, that is, the right edge of the SA is shifted to the right; otherwise, the width of the SA is increased to $\lfloor 1.4q \rfloor + u_i^{\mathrm{SA}_j} t$ towards the left. If $v_i^{\mathrm{SA}_j} t \geq 0$ the height of the SA is increased to $\lfloor 1.4q \rfloor + v_i^{\mathrm{SA}_j} t$ vertically down, that is, the bottom edge is shifted down; otherwise, it is increased by the same amount vertically upwards by shifting the top edge by $v_i^{\mathrm{SA}_j} t$.

Once SA is set, we use SSIM [36], an image quality assessment algorithm based on image formation, to detect the marker within each SA. In particular, SSIM [36] combines the luminance, contrast, and structure comparisons between a candidate block (always of the same size as the marker template block) and the marker template, and outputs a similarity value between 0 and 1. Using full motion search, we search for the candidate block within SA which has the maximum similarity value with the marker template. The centre coordinate of the found marker is the new marker position and is used to update the observation model of the DKF, $\hat{\mathbf{z}}_i$, which is in turn used to update the dynamic model, $\hat{\mathbf{s}}_i$.

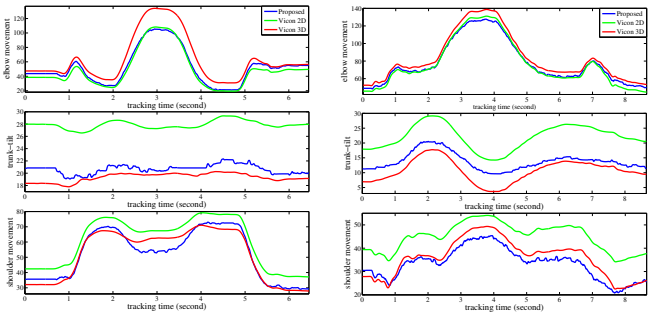
In summary, the advantage of the proposed approach comes from the proposed dynamic SA position and size update and marker detection via block matching using SSIM [36].

We note that only the waist marker can sometimes (rarely) be occluded, in which case we perform the same procedure as in [8]. For the upper limb motion analysis, the centre coordinates of pelvis, cervical spine, shoulder, elbow, and wrist markers obtained by marker tracking are next used for



(a) healthy subject (b) stroke survivor

Fig. 4: Marker trajectories.



(a) healthy subject (b) stroke survivor

Fig. 5: Automatically calculated joint angles (in degrees) on the upper limb motion. Top row: elbow movement α; middle row: trunk-tilt β; bottom row: shoulder movement γ.

visualization and autonomous joint angle calculation.

B. Autonomous joint angle calculation and visualization

During the tracking process, three joint angles - elbow movement α, trunk-tilt β, and shoulder movement γ, are, automatically and in real time, calculated on a frame-by-frame basis according to the centre coordinates of the detected markers. We record the marker trajectories by mapping the centre coordinates of all detected markers into a single frame. By working with practitioners and taking their feedback, we design a user interface in order to visualize all marker trajectories, and joint angles and check accuracy w.r.t benchmarks, as shown in Fig. 3. Via the interface, one can choose the video to be processed, and select (reselect if needed) the marker templates by mouse-click on the video frame shown in the "Current frame" panel. The "Template" panel then displays the appearance and centre coordinates of the marker templates. The marker tracking process begins by clicking "Start tracking", followed by showing appearance of the detected marker blocks in the "Tracking" panel and marker trajectories and joint angles, where Vicon 3D is the original tracking result from the Vicon system and Vicon 2D projects the 3D result to one of the three orthogonal Vicon system planes that is closely parallel to the plane of camera scene [8] in the "Result" panel.

Fig. 4 shows the marker trajectories of one trial from a healthy subject and one from a stroke survivor. The corresponding joint angles for these examples shown in Fig. 5 indicate that the joint angle plots of the proposed method closely follow those of the benchmarks Vicon 2D and 3D.

C. Subject classification

The aim of subject classification is to explicitly classify all participants into a healthy group and a patient group (binary classification) or a healthy group and several stroke groups with different levels of impairment [13] using the variations of the three tracked joint angles. Building on the principles of RGS [17], [20], we attempt to solve these binary and multi-class classification problems. Since binary classification is a special case of the multi-class classification, in the following we describe only the proposed multi-class classification schemes.

As classification features we use the standard deviation that is able to quantify the variation of a joint angle over one trial.

RGS is achieved by constructing a graph signal — using vertices to represent data elements with weighted edges connecting these vertices, and then applying regularization on the constructed graph signal to find an updated signal with minimum variation [17], [20]. We propose three RGS multi-class classification methods: "one-against-one" (OAO−RGS) — classify two classes at a time and next use the voting strategy, suggested in [37], to designate the final class for each sample, "one-against-all" (OAA−RGS) — consider one class at a time and group the other classes into a single class, and "once-for-all" (OFA−RGS) — classify all classes at once.

For OAO−RGS, we first design $l(l-1)/2$ binary classifiers, where $l > 2$ is the number of classes. Each classifier is trained using data from two of the l classes. In particular, given a set of data from Classes a and b: $\{\mathbf{x}_i^{ab}, y_i\}, y_i \in \{+1, 0, -1\}, \mathbf{x}_i^{ab} \in \mathbb{R}^V, i = 1, \ldots, D$, where all data elements with known labels construct the set of two-class training data: $\{\mathbf{x}_i^{ab}, y_i\}, y_i \in \{+1, -1\}, \mathbf{x}_i^{ab} \in \mathbb{R}^V, i = 1, \ldots, N, N < D$, where D and N are the total number of samples and the number of training samples, respectively. For the classifier on data from Classes a and b, we define a connected, undirected, and weighted graph $G^{ab} = (\mathcal{X}^{ab}, \zeta^{ab}, \mathbf{J}^{ab})$, where $\mathcal{X}^{ab} = \{\mathcal{X}_1^{ab}, ..., \mathcal{X}_D^{ab}\}$ is a set of vertices corresponding to dataset $\mathbf{x}^{ab} = \{\mathbf{x}_i^{ab}, \ldots, \mathbf{x}_D^{ab}\}$, ζ^{ab} denotes a set of edges, and $\mathbf{J}^{ab}$ denotes a weighted adjacency matrix. In particular, the weight $\mathbf{J}_{i,j}^{ab}$ on edge $\zeta_{i,j}^{ab}$ indicates the graph similarity of vertices $\mathcal{X}_i^{ab}$ and $\mathcal{X}_j^{ab}$, and is commonly defined by a Gaussian weighting function as:

$$\mathbf{J}_{i,j}^{ab} = \begin{cases} \exp(-\frac{\left\|\mathbf{x}_i^{ab} - \mathbf{x}_j^{ab}\right\|_2^2}{2\theta^2}) & \text{if } \left\|\mathbf{x}_i^{ab} - \mathbf{x}_j^{ab}\right\|_2^2 \leq \tau, \\ 0 & \text{otherwise,} \end{cases} \quad (1)$$

where θ denotes the Gaussian standard deviation, and τ is a threshold on the squared Euclidean distance of two vertices $\mathcal{X}_i^{ab}$ and $\mathcal{X}_j^{ab}$. Furthermore, we define a mapping of the graph G^{ab} as follows: $\mathbf{h}^{ab}$: $\mathcal{X}^{ab} \to \mathbb{R}$, $\mathcal{X}_n^{ab} \mapsto h_n^{ab}$, or $\mathbf{h}^{ab} = (h_1^{ab}, \ldots, h_D^{ab})^T \in \mathbb{R}^D$, where h_i^{ab} corresponds to vertex $\mathcal{X}_i^{ab}$ and data element $\mathbf{x}_i^{ab}$, and is given by: $h_i^{ab} = 1$ if $\mathcal{X}_i^{ab}$ belongs to Class a, -1 if $\mathcal{X}_i^{ab}$ belongs to Class b, and 0 if class is unknown.

Next, as in [17], we use the total variation on a graph (TV_G) to measure the total variation of G^{ab}:

$$\mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab}) = \frac{1}{\|\mathbf{h}^{ab}\|_2^2} \left\| \mathbf{h}^{ab} - \frac{1}{|\eta_{max}^{ab}|} \mathbf{J}^{ab} \mathbf{h}^{ab} \right\|_2^2 \quad (2)$$

where the product $\tilde{\mathbf{h}}^{ab} = \mathbf{J}^{ab}\mathbf{h}^{ab}$ is the output of the *graph shift* [17], a nontrivial graph filter; η_{max}^{ab} is an eigenvalue of $\mathbf{J}^{ab}$ that has the largest amplitude with constraint $|\eta_{max}^{ab}| \geq |\eta_i^{ab}|, 1 \leq i \leq D$. The objective of the classification

on $\mathrm{TV}_{G^{ab}}$ is to update all unknown labels within $\mathbf{h}^{ab}$ to get the lowest total variation on a graph [20], that is, a minimum $\mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab})$: $\mathbf{h}^{ab'} = \arg\min_{\mathbf{h}^{ab}\in\mathbb{R}^D} \mathrm{TV}_{G^{ab}}(\mathbf{h}^{ab})$.

We apply the above OAO−RGS classification procedure using all $l(l-1)/2$ binary RGS-based classifiers and use the voting strategy of [37] to designate classes.

For OAA−RGS, we design l binary classifiers. Each classifier is for data from one of the l classes and the group of remaining $l-1$ classes. In particular, we follow the procedure on graph construction as above, and define a graph G^{all} for data from all l classes. We then defined l different $\mathbf{h}$'s, i.e., l different mappings of the same graph G^{all}, for data from each of the l classes, and minimize each corresponding $\mathrm{TV}_{G^{all}}(\mathbf{h})$ to designate the class labels for each set of testing samples.

For OFA−RGS, we adopt the same graph G^{all} as used in OAA−RGS. Instead of using the binary mapping $\mathbf{h}^{ab}$, we define a multi-class graph mapping $\mathbf{h}^{all}$ for G^{all} (see Section III-C). We then minimize the total variation on G^{all}, that is, to get a minimum $\mathrm{TV}_{G^{all}}(\mathbf{h}^{all})$ and designate the class labels.

We discuss the multi-class classification process on the targeted upper limb motion analysis, and evaluate the performance of above three RGS methods, in Section III-C.

III. Experimental Results

In this section, we report the following experimental results:

- Comparison of bullseye marker tracking performance of the proposed DKF-SSIM tracking with four benchmark tracking methods JCTH [9], TLD [10], STR [22], and DKF-SSIM without the SA update (DKF-SSIM WSA).
- Separate validation of the proposed system with Vicon 2D and Vicon 3D (see Section II-B) for the group of healthy subjects and the group of stroke survivors since the stochastic movements of the stroke survivors make tracking more challenging.
- Evaluation of binary and OAO−, OAA− and OFA−RGS multi-class classification methods (Section II-C) for classifying all subjects into healthy and stroke groups.

Each video is captured using a digital camera EX-FH20 EXILIM (Casio Computer Co., Ltd., Tokyo, Japan) with 360×480 resolution. We adapt the camera calibration method from [38], where the coefficients of the radial distortion are obtained by solving a nonlinear minimization problem with the Levenberg-Marquardt Algorithm [39], to correct lens distortion of the acquired video frames before marker tracking. For benchmarking and validation, we simultaneously capture video with the 12-camera Vicon MX Giganet 6×T40 and 6×T160 (Vicon Motion Systems Ltd., Oxford, UK) optical motion analysis system (100fps), that is recognised as the state of the art [5] and commonly used in clinical rehabilitation practice. Fig. 1(a) shows a sample frame, where one out of the 12 Vicon infrared cameras is highlighted by a red square.

The proposed system is validated on 10 participants, including 5 healthy subjects and 5 stroke survivors. Each of the 10 participants performed 5 RTG trials, i.e., a total of 50 video clips are used, with a frame rate of 100fps for fair comparison with Vicon. The size of each marker template is always $q \times q = 11 \times 11$ pixels, which was heuristically found for optimal appearance representation of each marker that results in best tracking accuracy without sacrificing much computation cost.

TABLE I: Bullseye marker tracking on healthy subjects.

Method	Precision	Recall	PMR
JCTH [9]	0.581	0.581	27.5%
TLD [10]	0.958	0.922	64.6%
STR [22]	0.974	0.974	80.3%
DKF-SSIM WSA	0.852	0.852	82.8%
DKF-SSIM	0.998	0.998	97.3%

TABLE II: Bullseye marker tracking on stroke survivors.

Method	Precision	Recall	PMR
JCTH [9]	0.507	0.507	16.9%
TLD [10]	0.913	0.894	52.4%
STR [22]	0.955	0.955	81.7%
DKF-SSIM WSA	0.781	0.781	75.2%
DKF-SSIM	0.980	0.980	94.6%

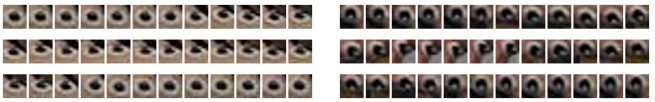
A. Bullseye Marker Tracking Performance Comparison

We randomly choose 1 of 5 trials for each participant, and select bullseye marker templates from the first frame of the corresponding video clip. Next, for each marker, we manually label the marker blocks in all frames of the video clip, with the same size as the marker template, as the ground truth (GT) to assess the bullseye marker tracking performance of all five methods. In the DKF-SSIM WSA approach, for each marker, we fix the size of SA at $\lfloor 1.4q \rfloor \times \lfloor 1.4q \rfloor$ and let the centre coordinate of the SA in the current frame be equal to the coordinate of the centre of the same marker detected in the previous frame.

We assess the performance by assigning True Positive (TP) if the detected marker block overlaps no less than 40% of the corresponding GT, and assigning False Positive (FP) otherwise. Furthermore, we define that a Perfectly Detected Marker (PDM) is assigned if the detected marker block overlaps no less than 90% of the corresponding GT. Let F be the total number of frames. Then, we define Precision=TP/(TP+FP), Recall=TP/F, and Perfect Marker Rate (PMR)={total number of PDMs}/F, where Precision and Recall indicate time proportion a tracking algorithm tracks the targeted marker; PMR indicates the accuracy of detecting the centre coordinate of the marker block.

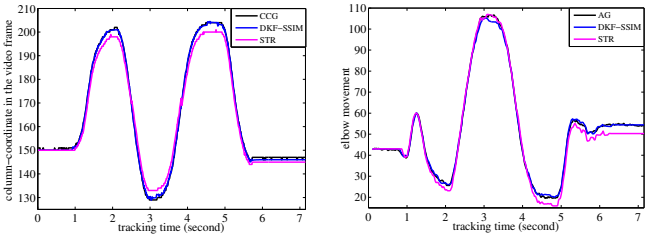
Tables I and II show the performance of the five tracking algorithms for bullseye marker tracking on healthy subjects and stroke survivors, respectively. JCTH [9] cannot recover from tracking failure caused by the *object-on-object* problem (see Section I). TLD [10] updates the marker model to help recover from the tracking failure, resulting in much higher scores than JCTH [9]. STR [22] outperforms TLD [10], but still cannot get marker centre accurately during out-of-plane rotation which commonly occurs when performing the RTG movement (see Fig. 6 for an illustration of the hand-labelled groundtruth shoulder and wrist markers over one trial).

The results also show that the SA update in each frame brings a 15-20% improvement in PMR, at the cost of a higher tracking complexity. Indeed, the average tracking and processing time per frame was 35msec and 43msec, for DKF-SSIM WSA and the proposed DKF-SSIM, respectively, measured in Matlab R2013b on a laptop running Windows 8.1, with Core i7 2820QM 2.3GHz processor and 16GB RAM.



(a) shoulder marker (b) wrist marker

Fig. 6: Hand-labelled groundtruth shoulder and wrist markers.



(a) wrist marker CC. (b) elbow movement (degree).

Fig. 7: Illustration of the tracking performance of the proposed DKF-SSIM and STR [22]. CC=column-coordinate. CCG=column-coordinate groundtruth. AG=angle groundtruth.

The proposed DKF-SSIM tracking-by-detection scheme is best suited for bullseye marker tracking due to its ability to incorporate dynamic and measurement models during tracking and combining the luminance, contrast, and structure features of the marker for detection. Since the position of the centre coordinate of the detected marker block has significant influence on the accuracy of the joint angle calculation, none of the four benchmark tracking methods are suited for autonomous joint angle calculation due to their resulting low PMR. To further demonstrate this, we show the tracking performance of the proposed DKF-SSIM and STR [22], the best benchmarking scheme among JCTH [9], TLD [10] and STR [22] according to Tables I and II, on one trial of a healthy subject in Fig. 7, where Fig. 7(a) shows the column-coordinate of the wrist marker given the benchmarking hand-labelled column-coordinate groundtruth, and Fig. 7(b) shows the corresponding elbow movement angle (degree) given the benchmarking angle groundtruth calculated from the hand-labelled groundtruth shoulder, elbow, and wrist markers. The corresponding error is shown in Table III.

TABLE III: Tracking error in Fig. 7. CC=column-coordinate.

	wrist marker CC (pixel)		elbow movement (degree)	
	mean error	max error	mean error	max error
STR [22]	2.6028	5	2.1291	5.4342
DKF-SSIM	0.5670	2.4215	0.7350	3.3349

B. Angle Accuracy Validation

We validate the proposed DKF-SSIM tracking with Vicon 2D and 3D using Bland-Altman plot [21] for evaluation of the limits of agreement. Bland-Altman plot is a typical clinical measurement scheme to evaluate a new measurement system based on an established one. In particular, let vectors $\mathbf{Q}_1$ and $\mathbf{Q}_2$ contain all measurements from Methods 1 and 2, respectively. For each value $\mathrm{Q}_1(i) \in \mathbf{Q}_1$ and corresponding $\mathrm{Q}_2(i) \in \mathbf{Q}_2$, Bland-Altman plot is constructed by assigning $[\mathrm{Q}_1(i) + \mathrm{Q}_2(i)]/2$ as the abscissa value, and $\mathrm{Q}_1(i) - \mathrm{Q}_2(i)$ as the ordinate value. Next, we calculate the mean difference (MD) and the standard deviation of $\mathbf{Q}_1$ and $\mathbf{Q}_2$, followed by

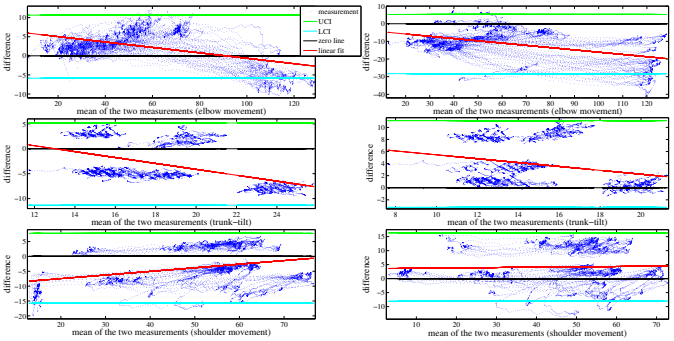


Fig. 8: Bland-Altman plots (in degrees) of all healthy subjects. Left column: P vs. V2. Right column: P vs V3. Top row: elbow movement α; middle row: trunk-tilt β; bottom row: shoulder movement γ.

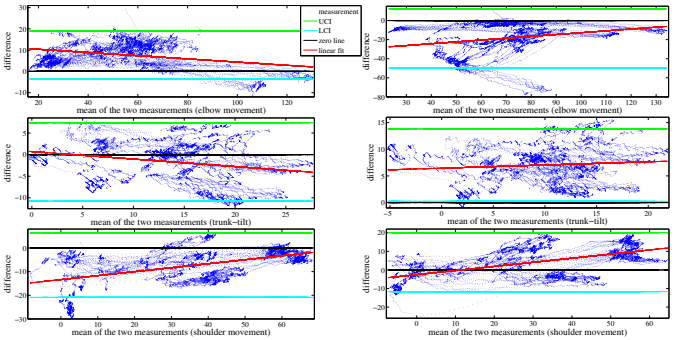


Fig. 9: Bland-Altman plots (in degrees) of all stroke survivors.

lower and upper 95% confidence interval (LCI, UCI) and a linear fit, all of which are based on the constructed Bland-Altman plot, for complete limits of agreement evaluation.

The dataset used contains 25 trials from healthy subjects and another 25 trials from stroke survivors. We group all 25-trial results of healthy subjects (stroke survivors) together forming three vectors $\mathbf{Q}_{\mathrm{X}}^{\alpha}$, $\mathbf{Q}_{\mathrm{X}}^{\beta}$, and $\mathbf{Q}_{\mathrm{X}}^{\gamma}$, where $\mathrm{X} = \{\mathrm{P}, \mathrm{V2}, \mathrm{V3}\}$, denotes (P)roposed, Vicon 2D (V2) or 3D (V3).

Figs 8 and 9 show the Bland-Altman plots based on above construction process for the healthy subjects and stroke survivors, respectively. Table IV shows the corresponding limits of agreement (LOA). Note that good LOA is indicated by small MD, narrow 95% CI, and a linear fit that is close to zero [21]. Since the deviation between the elbow movement α plane and camera scene plane (CSP) is more notable than that between the trunk-tilt β plane and CSP and that between the shoulder movement γ plane and CSP, validation of P and V3 on α shows a relatively large MD and wide 95% CI. Otherwise, P and V3 show good LOA on β and γ; P and V2 show good LOA for all motion patterns. In general, 3D information is needed in diagnostic systems. However, the above validation incorporates loss of 3D information, indicating that 2D suffices for the targeted RTG sagittal movement analysis. This is in accordance to the prior literature [40].

C. Subject Classification

As classification features we use the standard deviation of all three joint angles over one trial. That is, each data sample $(\sigma_{\alpha_i}, \sigma_{\beta_i}, \sigma_{\gamma_i})$ is a 3-dimensional feature vector that contains standard deviations of the joint angles α, β, and γ, where σ_{α_i}, σ_{β_i}, and σ_{γ_i} are the standard deviations during

TABLE IV: Limits of agreement (in degrees) between P and V2, and between P and V3 for all participants.

		Healthy subjects			Stroke survivors		
		MD	LCI	UCI	MD	LCI	UCI
P vs V2	α	2.38	-5.86	10.6	7.72	-3.51	19.0
	β	-3.08	-11.3	5.16	-1.68	-10.8	7.39
	γ	-4.02	-15.8	7.72	-7.26	-20.9	6.37
P vs V3	α	**-11.5**	**-28.3**	**5.24**	**-18.8**	**-49.8**	**12.3**
	β	3.93	-3.24	11.1	7.01	0.22	13.8
	γ	4.13	-8.07	16.4	4.22	-11.6	20.1

TABLE V: Levels of impairment of stroke survivors.

Stroke survivor	SS 1	SS 2	SS 3	SS 4	SS 5
Ordinal scale	2	5	1	4	2

one trial of angles α, β, and γ, respectively. We evaluate the performance of the classification algorithms under different sizes of the training and testing data by using following metric: Classification Accuracy $=$ {Number of correctly classified samples}/{Number of testing samples}.

First, we perform binary classification, whose task is to group all subjects into two groups: healthy and stroke patients. We compare the proposed `RGS` binary classifier to that of linear and non-linear (we use a Gaussian Radial Basis Function (`rbf`) kernel with scaling factor $\rho = 1$) SVM binary classifiers, denoted as `l-SVM` and `rbf-SVM`, respectively. The results are given in Fig. 10 expressed as Classification Accuracy. In particular, we assume that between 4% and 80% of randomly selected labels are known for training, perform 10,000 tests, and then get the averaged result. It can be seen that `RGS` shows competitive performance with `l-SVM` when the percentage of known labels is above 40% at lower complexity.

Next, we turn to the multi-class classification, whose task is to classify further patients into different recovery levels. Table V shows the levels of upper limb impairment for 5 stroke participants, reported from a recruited rater, a biomechanics researcher with over ten years of experience in biomechanics data analysis, by observational assessment [13], [41]. Thus, we define $l = 5$ classes for all experimental data: Healthy, Stroke with ordinal scale 1 (OS 1), OS 2, OS 4, and OS 5, denoted as Class q, $q = 1, ..., 5$, respectively.

For `OAO-RGS`, we design $l(l-1)/2$ binary classifiers. For each classifier, we first define a graph for data from two of the l classes: a connected, undirected, and weighted graph $G = (\mathcal{X}, \zeta, \mathbf{J})$, with vertices $\mathcal{X} = \{\mathcal{X}_1, \ldots, \mathcal{X}_D\}$ correspond to the dataset $\mathbf{x} = \{\mathbf{x}_1, \ldots, \mathbf{x}_D\}$, edges ζ, and a weighted adjacency matrix $\mathbf{J}$ defined using (1), with $\theta = 1$ and $\tau = 100$ which balances the number of non-zero entries in $\mathbf{J}$ and computation time, where $\mathbf{x}_i = (\sigma_{\alpha_i}, \sigma_{\beta_i}, \sigma_{\gamma_i})$. Next, we define $\mathbf{h}$, i.e., the mapping of the graph G, and minimized TV_G of G as defined in (2). Finally, we use the voting strategy [37] to designate groups for all testing data.

For `OAA-RGS`, we design l binary classifiers. For each classifier, we first define a graph G^{all} for data from all l classes with the same parameter setting $\theta = 1$ and $\tau = 100$, for $\mathbf{J}$, then defined $\mathbf{h}$ for G^{all}, followed by minimization of each corresponding $\mathrm{TV}_{G^{all}}(\mathbf{h})$ and the voting strategy [37] to designate the class labels for each set of testing samples.

For `OFA-RGS`, we apply the same graph G^{all} as used in `OAA-RGS`, and defined a multi-class graph mapping $\mathbf{h}^{all}$ of G^{all} as follows: $h_i^{all} = -7 + 2q$ if $\mathcal{X}_i$ belongs to Class $q, q = 1, ..., 5$, and 0 if class is unknown.

We then perform $\mathbf{h}^{all'} = \arg\min_{\mathbf{h}^{all} \in \mathbb{R}^D} \mathrm{TV}_{G^{all}}(\mathbf{h}^{all})$ for class labels of all testing samples.

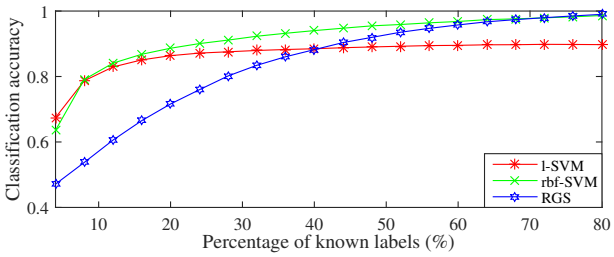


Fig. 10: Binary classification accuracy of testing data.

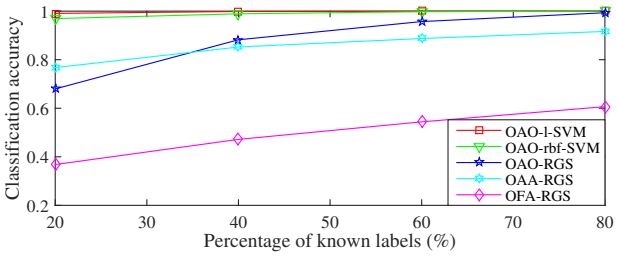


Fig. 11: Multi-class classification accuracy of testing data.

For benchmarking, we adopt "one-against-one" multi-class `SVM` classification [37], [42], [43], a competitive approach among five multi-class `SVM` classification methods compared in [44]. We first train $l(l-1)/2$ binary linear / non-linear (we use `rbf` kernels with scaling factor $\rho = 1$ which gives best classification results without overfitting) `SVM` classifiers, and then classify all testing data by using voting strategy in [37], denoted as `OAO-l-SVM` and `OAO-rbf-SVM`, respectively.

We evaluate the above 5 multi-class classification methods using k-fold cross-validation adapted from [45]. In particular, we set $k = 5$, i.e., 4 folds are used for training and the last fold is used for evaluation. We repeat this process k times, leaving one different fold for evaluation each time. The ith process outputs a confusion matrix of data counts, denoted as $\mathbf{C}_c^i = [c_{11}^i, ..., c_{1l}^i; ...; c_{l1}^i, ..., c_{ll}^i]$, whose columns represent the classifier prediction, and rows represent the true classes, e.g., the value of index c_{ij}^i in $\mathbf{C}_c^i$ increases by 1 if a data sample that belongs to Class i is classified as Class j. k-fold cross-validation finally combines all $\mathbf{C}_c^i$'s into a single confusion matrix of data counts $\mathbf{C}_c$ with indices $c_{ij} = \sum_{i=1}^k c_{ij}^i$, and outputs the corresponding accuracy (**acc**) given by: **acc** $= \sum_{i=1}^k c_{ii} / \sum_{i=1}^k \sum_{j=1}^k c_{ij}$. Note that $\mathbf{C}_c$ can be alternatively represented as a confusion matrix of recognition rates, denoted as $\mathbf{C}_r = [\mathbf{C}_c(1,:)/\sum \mathbf{C}_c(1,:); ...; \mathbf{C}_c(k,:)/\sum \mathbf{C}_c(k,:)]$.

Next, we show the evaluation result of the above 5 multi-class classification methods using Accuracy in Fig. 11 (averaged over 10,000 runs based on the assumption that between 20% and 80% of randomly selected labels are known for training) and k-fold cross-validation in Table VI, where $\overline{t_r}$ and $\overline{t_e}$ denote the average execution time for training and testing during the ith process of k-fold cross-validation, respectively. `OFA-RGS` is not competitive with any of above 4 methods. The performance of `OAO-RGS` is between `SVM` methods and `OAA-RGS` when the percentage of known labels is above 40%. `SVM` methods and `OAO-RGS` achieve the highest **acc**, where

TABLE VI: k-fold cross-validation result.

Method	$\mathbf{C}_r$	acc	$\overline{t_r}$ (ms)	$\overline{t_e}$ (ms)
OAO-1-SVM	$\mathbf{I}_5$	**1**	58.3	3.9
OAO-rbf-SVM	$\mathbf{I}_5$	**1**	53.1	4.6
OAO-RGS	$\mathbf{I}_5$	**1**	22.5	3.3
OAA-RGS	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0.1 & 0 & 0.9 & 0 & 0 \\ 0.2 & 0 & 0 & 0.8 & 0 \\ 0.2 & 0 & 0 & 0 & 0.8 \end{bmatrix}$	**0.94**	6.5	3.7
OFA-RGS	$\begin{bmatrix} 0.68 & 0.08 & 0.24 & 0 & 0 \\ 0 & 0.4 & 0.6 & 0 & 0 \\ 0 & 0 & 0.9 & 0.1 & 0 \\ 0 & 0 & 0.2 & 0.8 & 0 \\ 0 & 0 & 0.2 & 0.8 & 0 \end{bmatrix}$	**0.64**	7	1.5

OAO−RGS performs faster than both SVM methods. Indeed, OAO−RGS performs over 100% and 15% faster, for training and testing, respectively, than the SVM methods. The overall performance of OAO−RGS indicate that our decision support system has the potential to accurately classify participants into a healthy group and different stroke groups with the aid of levels of impairment [13].

We note that the above multi-class analysis is provided to demonstrate the potential of the proposed methods, since the amount of data is insufficient to make firm conclusions.

IV. Conclusion

Currently available optical motion analysis systems are expensive and require multiple infrared cameras, large laboratory space, and operational expertise to assess motor impairment of a stroke survivor. In this paper we propose and evaluate an alternative, portable, and cheap, single-camera decision support system with the following components: simultaneous multiple bullseye marker tracking, autonomous joint angle calculation, visualization, and subject classification. Validation of the proposed tracking method with the current state-of-the-art Vicon optical motion analysis system shows overall good limits of agreement on the upper limb motion analysis. In addition, we designed three RGS binary and multi-class classification methods, of which OAO−RGS has strong potential to explicitly classify participants into a healthy group and different stroke groups with the aid of levels of impairment. In practice, for a 10-second trial, a patient can get his/her upper limb kinematics assessed in under 2 minutes, given the average processing time (see Section III-A) per video frame. Experimental results show that the proposed decision support system can track the markers with high accuracy, capture the upper limb motion explicitly, and give stroke survivors and clinicians visual and written feedback based on classification with the aid of impairment levels.

Acknowledgment

The authors are grateful to all the participants that contributed in the experiments that form the RTG dataset.

References

[1] C. Yang, A. Kerr, V. Stankovic, L. Stankovic, and P. Rowe, "Upper limb movement analysis via marker tracking with a single-camera system," in *IEEE Intl. Conf. Image Processing*, Paris, France, Oct. 2014.

[2] A. Kerr, C. Yang, V. Stankovic, and P. Rowe, "Accuracy of a 2D video system for measuring upper limb movement in stroke survivors," in *World Stroke Congress*, Istanbul, Turkey, Oct. 2014.

[3] R. L. Sacco et al., "An updated definition of stroke for the 21st century," *Stroke*, vol. 44, pp. 2064-2089, Jul. 2013.

[4] P. Langhorne, F. Coupar, and A. Pollock, "Motor recovery after stroke: A systematic review," *The Lancet Neurology*, vol. 8, pp. 741-754, Aug. 2009.

[5] J. G. Richards, "The measurement of human motion: A comparison of commercially available systems," *Human Movement Science*, vol. 18, pp. 589-602, Oct. 1999.

[6] J. K. Aggarwal and Q. Cai, "Human Motion Analysis: A Review," *Computer Vision and Image Understanding*, vol. 73, no. 3, pp. 428-440, Mar. 1999.

[7] R. Poppe, "Vision-based human motion analysis: An overview," *Computer Vision and Image Understanding*, vol. 108, no. 1-2, pp. 4-18, Oct. 2007.

[8] C. Yang, U. Ugbolue, B. Carse, V. Stankovic, L. Stankovic, and P. Rowe, "Multiple marker tracking in a single-camera system for gait analysis," in *IEEE Intl. Conf. Image Processing*, Melbourne, Australia, Sep. 2013.

[9] J. Ning, L. Zhang, D. Zhang, and C. Wu, "Robust object tracking using joint color-texture histogram," *Int. J. Pattern Recognition and Artificial Intelligence*, vol. 23, pp. 1245-1263, Nov. 2009.

[10] Z. Kalal, K. Mikolajczyk, and J. Matas, "Tracking-learning-detection," *IEEE TPAMI*, vol. 34, pp. 1409-1422, Jul. 2012.

[11] B. H. Dobkin, "Rehabilitation after Stroke," *New England Journal of Medicine*, vol. 352, no. 16, pp. 1677-1684, April 2005.

[12] H. M. Clayton and H. C. Schamhardt, "Measurement techniques for gait analysis," *Equine locomotion*, pp. 55-76, Mar. 2001.

[13] M. K. Hayes et al., "The american heart association stroke outcome classification: Executive summary," *Stroke*, vol. 97, pp. 2474-2478, Jun. 1998.

[14] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE Signal Proc. Magazine*, vol. 30, no. 3, pp. 83-98, May 2013.

[15] J. Xu et al., "Graph-based topic-focused retrieval in distributed camera network," *IEEE Trans. Multimedia*, vol. 15, no. 8, pp. 2046-2057, Dec. 2013.

[16] B. Macchiavello et al., "Loss-resilient coding of texture and depth for free-viewpoint video conferencing," *IEEE Trans. Multimedia*, vol. 16, no. 3, pp. 711-725, Apr. 2014.

[17] A. Sandryhaila and J. M. F. Moura, "Classification via regularization on graphs," in *IEEE GlobalSIP*, Austin, TX, Dec. 2013.

[18] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, pp. 273-297, Sep. 1995.

[19] C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, pp. 121-167, Jun. 1998.

[20] A. Sandryhaila and J. M. F. Moura, "Discrete signal processing on graphs: Frequency analysis," *IEEE Trans. Signal Proc.*, vol. 62, pp. 3042-3054, Jun. 2014.

[21] J. M. Bland and D. G. Altman, "Statistical methods for assessing agreement between two methods of clinical measurement," *The Lancet*, vol. i, pp. 307-310, Feb. 1986.

[22] S. Hare, A. Saffari, and P. H. S. Torr, "Struck: Structured output tracking with kernels," in *IEEE ICCV*, Barcelona, Spain, Nov. 2011.

[23] A. Yurton et al., "Home-based reach-to-grasp training for people after stroke: study protocol for a feasibility randomized controlled trial," *Trials*, vol. 14, no. 1, Apr. 2013.

[24] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: A survey," *ACM Computing Surveys*, vol. 38, Dec. 2006.

[25] Y. Wu, J. Lim, and M.-H. Yang, "Online object tracking: A benchmark," in *IEEE CVPR*, Portland, OR, Jun. 2013.

[26] X. Li et al., "A survey of appearance models in visual object tracking," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 4, pp. 58:1–58:48, Oct. 2013.

[27] A. W. M. Smeulders et al., "Visual tracking: An experimental survey," *IEEE TPAMI*, vol. 36, no. 7, pp. 1442–1468, Jul. 2014.

[28] T. J. Broida and R. Chellappa, "Estimation of object motion parameters from noisy images," *IEEE TPAMI*, vol. 8, pp. 90-99, Jan. 1986.

[29] A. Yilmaz, X. Li, and M. Shah, "Contour-based object tracking with occlusion handling in video acquired using mobile cameras," *IEEE TPAMI*, vol. 16, pp. 1531-1536, Nov. 2004.

[30] S. Zhang et al., "Object tracking with multi-view support vector machines," *IEEE Trans. Multimedia*, vol. 17, no. 3, pp. 265-278, Mar. 2015.

[31] Y. Yuan et al., "Visual object tracking by structure complexity coefficients," *IEEE Trans. Multimedia*, vol. 17, no. 8, pp. 1125-1136, Aug. 2015.

[32] X. Zhang, W. Hu, W. Qu, and S. Maybank, "Multiple object tracking via species-based particle swarm optimization," *IEEE TCSVT*, vol. 20, pp. 1590-1602, Nov. 2010.

[33] Z. H. Khan, I. Y. H. Gu, and A. G. Backhouse, "Robust visual object tracking using multi-mode anisotropic mean shift and particle filters," *IEEE TCSVT*, vol. 21, pp. 74-87, Jan. 2011.

[34] G. Welch and G. Bishop, "An introduction to the Kalman Filter," in *Course 8 of SIGGRAPH 2001*, LA, Aug. 2001.

[35] R. E. Kalman, “A new approach to linear filtering and prediction problems,” *Transaction of the ASME-Journal of Basic Engineering*, vol. 82, pp.35-45, Mar. 1960.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, “Image quality assessment: from error visibility to structural similarity,” *IEEE Trans. Image Proc.*, vol. 13, pp. 600-612, Apr. 2004.

[37] J. H. Friedman. (1996) Another approach to polychotomous classification. Tech. Rep., Dept. Statistics, Stanford University, Stanford, CA.

[38] Z. Zhang, “A flexible new technique for camera calibration,” *IEEE TPAMI*, vol. 22, pp. 1330-1334, Nov. 2000.

[39] J. More, “The levenberg-marquardt algorithm: implementation and theory,” in *Numerical Analysis*, G. A. Watson, Ed. Berlin: Springer, 1977, pp. 105-116.

[40] U. C. Ugbolue et al., “The evaluation of an inexpensive, 2D, video based gait assessment system for clinical use,” *Gait & Posture*, vol.38, pp. 483-489, Jul. 2013.

[41] J. Bernhardt et al., “Accuracy of observational kinematic assessment of upper-limb movements,” *Physical Therapy*, vol.78, pp. 259-270, Mar. 1998.

[42] U. H.-G. Kressel, “Pairwise classification and support vector machines,” in *Advances in Kernel Methods-Support Vector Learning*, B. Schölkopf, C. J. C. Burges, and A. J. Smola, Eds. Cambridge, MA: MIT Press, 1999, pp. 255-268.

[43] C.-C. Chang and C.-J. Lin, “LIBSVM: A library for support vector machines,” *ACM Trans. Intel. Systems and Techn.*, vol. 2, no. 3, pp. 1-27, Apr. 2011.

[44] C.-W. Hsu and C.-J. Lin, “A comparison of methods for multiclass support vector machines,” *IEEE Trans. Neural Networks*, vol. 13, no. 2, pp. 415-425, Mar. 2002.

[45] R. Kohavi, “A study of cross-validation and bootstrap for accuracy estimation and model selection,” in *Intl. Joint Conf. Artificial Intelligence*, Montreal, Quebec, Canada, 1995.