<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper published in *Computers & graphics*. This paper has been peer-reviewed but does not include the final publisher proof-corrections or journal pagination.

Citation for the original published paper (version of record):

Alexanderson, S., O'Sullivan, C., Beskow, J. (2017)
Real-time labeling of non-rigid motion capture marker sets.
Computers & graphics, 69(Supplement C): 59-67
<https://doi.org/10.1016/j.cag.2017.10.001>

Access to the published version may require subscription.

N.B. When citing this work, cite the original published paper.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-218254>

Special Section on Motion in Games 2016

Real-time labeling of non-rigid motion capture marker sets

Simon Alexanderson^{a,*}, Carol O'Sullivan^b, Jonas Beskow^a^a KTH - Speech, Music and Hearing, Lindsetdtsv. 24, Stockholm 10044, Sweden^b Trinity College Dublin, College Green, Dublin 1, Ireland

ARTICLE INFO

Article history:

Received 1 August 2017

Revised 1 October 2017

Accepted 2 October 2017

Available online 12 October 2017

Keywords:

Animation

Motion capture

Hand capture

Labeling

ABSTRACT

Passive optical motion capture is one of the predominant technologies for capturing high fidelity human motion, and is a workhorse in a large number of areas such as bio-mechanics, film and video games. While most state-of-the-art systems can automatically identify and track markers on the larger parts of the human body, the markers attached to the fingers and face provide unique challenges and usually require extensive manual cleanup. In this work we present a robust online method for identification and tracking of passive motion capture markers attached to non-rigid structures. The method is especially suited for large capture volumes and sparse marker sets. Once trained, our system can automatically initialize and track the markers, and the subject may exit and enter the capture volume at will. By using multiple assignment hypotheses and soft decisions, it can robustly recover from a difficult situation with many simultaneous occlusions and false observations (ghost markers). In three experiments, we evaluate the method for labeling a variety of marker configurations for finger and facial capture. We also compare the results with two of the most widely used motion capture platforms: Motion Analysis Cortex and Vicon Blade. The results show that our method is better at attaining correct marker labels and is especially beneficial for real-time applications.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

Optical marker-based motion capture is a mature and dominant technology for capturing detailed human motion in many areas such as bio-mechanics, film and video games. The technology provides many desirable features such as high accuracy and sampling rates and can be used as a single means to capture body and finger motion as well as facial expression. Among the main challenges for optical motion capture using passive markers is the identification and tracking of the markers, commonly referred to as labeling. The difficulties arise due to the fact that the markers look identical from the point of view of the system and their identities need to be inferred from structural cues or tracked over time, something that is further challenging in cases of severe occlusions.

Current state-of-the-art motion capture systems can reliably label markers on the larger parts of the human body, also in large capture volumes. However, markers on the more articulated body parts, such as the face and fingers, pose unique challenges and usually require extensive manual labelling. For facial capture, alternative markerless methods (such as video based tracking using head-mounted cameras) has gained in popularity, but this adds

cost and complexity to the setup, and there are still many domains in which head-mounted cameras are too intrusive to be used. For finger capturing, the only viable solutions for large volumes are to use either data gloves or sparse marker sets with optical motion capture, [1]. In our work, we connect to the recent advances in data-driven methods to produce high quality hand and finger animation from sparse marker sets [2–4], and address the problems of automatic labelling of such markers. Sparse marker sets prove to be especially challenging for existing labeling algorithms. This is mainly due to the fact that sparsity reduces the structural information available to the point where underlying skeleton models, commonly used in existing labeling algorithms, are difficult to apply.

In this paper, we present an extended version of our paper on robust algorithms for automatic labeling of finger markers, [5]. In addition to previously reported work, we show how our method can be extended to simultaneously label multiple marker sets in close interaction, and present new results of labeling face and finger markers in a full performance capture setup. We also show how our method integrates with data-driven methods for reconstructing full marker sets from sparse data, and hence allows users to reduce the number of markers in a capture without significant loss of quality.

At the core of our system is an algorithm to generate multiple assignment hypotheses based on the spatial distribution of

* Corresponding author.

E-mail address: simonalex@kth.se (S. Alexanderson).

Fig. 1. A selection of sparse marker sets for finger capture: (a) and (b) [6]; (c) Opti-track Motive; (d) [7,8]; (e) [3]; and (f) [9]. The top row shows common marker sets used in the industry, and the bottom row shows the recommended marker sets from the research community. Note the large marker separation in the top row, facilitating automatic labeling.

Fig. 2. Capture volume of 7 m × 12 m × 5 m.

the markers, and another algorithm to select the best sequence of assignments in time. A key characteristic of our method is the domain in which the assignment hypotheses are generated. While other methods generate assignments from the *temporal* domain, i.e. from the predicted marker positions at each frame, and use an initialization phase (usually involving a T-pose) to commence tracking, our method continuously generates a fixed set of assignment hypotheses from the *spatial* domain, and treats tracking as an optimization problem to find the most probable path through the hypothesis space. In this way, our method can continuously reinitialize the marker labels even after long occlusions. By using multiple assignment hypotheses, no hard decisions are made at times where the assignments are ambiguous due to occlusions and/or ghost markers, and the algorithm has a chance to correct errors as more evidence becomes available.

We evaluate our method in three experiments. The first experiment covers finger capturing using a variety of different marker sets described in the literature (see Fig. 1), and shows that our method is able to provide correct labels for over 99.6% of the data for all of the marker sets. The second experiment covers finger capturing in a large volume (see Fig. 2). Bench-marked against two of the most dominant commercial platforms, Motion Anal-

ysis Cortex¹, and Vicon Blade², our method is better at attaining correct marker labels in general and is particularly beneficial for fragmented data. The third experiment covers simultaneous labelling of face and finger markers in a full performance capture and demonstrates how the method is used in conjunction with data-driven methods to generate rich data sets from sparse markers.

As our method is working in real-time, it is of special use to the video-games and film industries, which require large capture volumes for in-game motion and cinematics, and real-time capabilities for Virtual Reality, Previs and Virtual Production.

2. Related work

Early marker labeling techniques emerge from the field of Multiple Target Tracking (MTT) [10], which was originally developed for tracking radar plots. One of the most successful MTT algorithms is Multiple Hypothesis Tracking [11], which allows for soft decision making when the observations are noisy and the tracking situation is ambiguous. A limitation of using MTT algorithms for motion capture is that they do not take structural information into account, and thus needs to be manually initialized at the first time frame as well as after longer periods of gaps. In most motion capture scenarios, the motions of the markers are correlated in some way, which may be exploited for labeling. Gennari et al. [12] integrate shape constraints in MTT, but do not initialize marker identities or use multiple hypotheses. Also Yu et al. [13] exploit structural information, but their algorithm requires a large number of markers and is not suitable for sparse, non-rigid marker sets.

Other studies focus on simultaneous labeling and skeleton solving using an underlying skeleton model. Ringer and Lasenby [14] developed a multiple hypotheses tracker and demonstrate their method on human body motion. Meyer et al. [15] used a probabilistic framework for automatic online labeling of full-body marker sets, and Schubert et al. [16] extend this method to be able to initialize the tracking using an arbitrary pose. As opposed to our approach, these methods require dense enough marker sets to uniquely define the pose of the underlying skeleton model. Our method is developed for sparse marker sets and data-driven pose estimation, where as few as 3 markers may be used to drive more than 20 degrees of freedom of finger motion. Recently, Maycock et al. [9] developed a labeling system using an inverse kinematics (IK) based skeleton, and demonstrated it for capturing hand and finger motion. However, their method requires a specialized initialization pose and does not use multiple hypotheses, and it is not clear how it would reinitialize in cases where several markers are occluded for longer time periods. In a study by Akhter et al. [17], a spatiotemporal model was developed to perform simultaneous labeling and gap-filling. The method was demonstrated on a dense set of 315 facial markers. However, in contrast to our domain where only a few loosely correlated markers exist, their data set contains a large amount of spatiotemporal correlation, making it possible to deduce lost marker positions from the trained model.

The capturing of hand motion is an active research field with many recent publications (see the state-of-the-art report [1] for an overview). While there have been major improvement in marker-less methods based on computer vision techniques and depth sensors, these methods still impose severe restrictions, e.g. on capture volumes, frame rates and tracking of parts that are in physical contact. According to [1], they are only appropriate in small volumes and have difficulties in reconstructing complex hand shapes. Other techniques exist based on instrumented gloves such as the

¹ <http://www.motionanalysis.com>.

² <http://www.vicon.com>.

Fig. 3. The three highest ranked assignment hypotheses ordered from left to right. The left hypothesis is correctly labeled..

Cyberglove³. These systems tend to be expensive and involve cumbersome and frequent calibration procedures, and do not deliver the same accuracy as marker based systems [1].

Motion capture for film and video games is usually performed in large studios where the actors can run, jump and perform different kinds of stunts. As large volumes require large markers (typically 10 mm) and reduced marker sets for the hands, there has been substantial research on methods to optimize animation quality from sparse marker configurations. The main objectives have been to find optimized marker layouts and to reconstruct full hand poses from previously recorded data. Proposed methods include using a combination of principal component analysis (PCA) and locally weighted regression (LWR) [2], mixture of factor analysis (MFA) clustering [4], and subspace-constrained inverse kinematics [3]. Hoyet et al. [18] investigate the *perceptual* difference between animated hand motion generated from a range of marker sets, and recommend a sparse marker set of 6 markers as good balance between manual post-processing efforts and animation quality. While these studies demonstrate the viability of marker based hand capture in large volumes, they do not address the labeling problem which is the focus in this paper. In Section 4.3 we use an alternative data-driven method derived from [19] to reconstruct marker data from sparse marker sets. This is done to show the viability of our capturing pipeline and we do not provide comparisons to the studies above.

3. Method

The underlying problem for passive marker labeling arises from a lack of individual discriminating features for identifying the markers. Instead, existing algorithms base the assignments on spatial inter-relations and temporal coherence. While markers placed on rigid objects or kinematic chains (such as human skeletons) provide structurally invariant features, marker placed on flexible structures such as fingers and faces yield much more ambiguous information. When using markers only on the fingertips, for example, several different assignments of marker labels may generate equally valid hand poses (see Fig. 3). The uncertainty in the spatial information is especially problematic if temporal coherence is deteriorated due to frequent occlusions or stretches of noisy data. Finger markers are particularly prone to occlusion when the fingers are flexed towards the ground or the body [20], and markers placed close to each other may be falsely reconstructed as one single marker, causing uncertainty in which trajectory the false marker should belong to.

To address these problems we base our algorithm on two core features: an assignment generation method for generating multiple ranked hypotheses from the spatial distribution of the unlabeled markers in each frame; and a hypothesis selection method for selecting a smooth sequence of assignments in time. By generating our assignments from the spatial domain rather than the temporal, we can automatically initialize the system after occlusions. By using multiple hypotheses, we can also handle ambiguous situations and postpone decisions until more discriminative observations arrive. Hypothesis generation uses a collection of Gaussian

Mixture Models (GMMs) to model each marker's location in space, while hypothesis selection uses Kalman filters [21] and the Viterbi algorithm [22] to determine the best sequence of hypotheses in time. These methods have the benefit of being fast and probabilistic, making them especially suitable for real-time applications. Our unoptimized Matlab prototype runs at 58 frames per s using 5 markers and 5 hypotheses per frame on a Intel Core i7 2.6 GHz laptop.

3.1. Hypotheses generation

A prerequisite for our method is that the unlabeled data is transformed to a local coordinate system following the marker set. This is achieved either by placing markers on the head or hand base forming a rigid structure or, if the algorithm is run in parallel with full-body capture, by providing the world-to-local transform from the skeleton solver. Given a marker set with M markers m_i , $i \in \{1, \dots, M\}$, we model the spatial distribution of each marker with a GMM, thus giving us a collection of M GMMs. At any frame t containing K unlabeled observations, the log likelihood L_{ij}^t of an unlabeled observation at position $\mathbf{y}_j^t$, $j \in \{1, \dots, K\}$, to be assigned marker label i is given by

$$L_{ij}^t(\mathbf{y}_j^t) = \log \left(\sum_l (w_{li} f(\mathbf{A}^t \mathbf{y}_j^t, \mu_{li}, \Sigma_{li})) \right) \quad (1)$$

where w_{li} , μ_{li} and Σ_{li} are the parameters for the l th mixture component of the GMM for marker m_i , $\mathbf{A}^t$ is the world-to-local transformation of the marker set, and $f(\mathbf{x}, \mu, \Sigma)$ is the multivariate Gaussian probability.

$$f(\mathbf{x}, \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp \left(-\frac{1}{2} (\mathbf{x} - \mu)^T \Sigma^{-1} (\mathbf{x} - \mu) \right) \quad (2)$$

The likelihoods from Eq. (1) form an $(M \times K)$ matrix $\mathbf{L}$ and the best assignment hypotheses can be found by solving the Linear Assignment Problem (LAP) using $\mathbf{L}$ as a reward matrix. LAP can be efficiently solved using several algorithms, see for example [23], and can be extended to find a ranked set of N best solutions using Murty's algorithm [24]. We store the N highest ranked assignments as our hypotheses, $\{\chi_1^t, \dots, \chi_N^t\}$ for each frame t , and use the log likelihood of each assignment hypothesis as the corresponding emission score.

A common situation in optical motion capture is the appearance of ghost markers. Ghost markers may occur from reflections from glossy objects in the volume or from errors in 3D point reconstruction, most notably for markers close to each other. As it is difficult for the labeler to distinguish a noisy observation from a simultaneous gap and a ghost, the algorithm needs to apply some tolerance threshold for an observation to be considered as a candidate marker. Generally, temporally based labelers do this by specifying a radius from the predicted positions according to the labeled markers' trajectories. This is however problematic for fragmented data as the predictions rapidly deteriorate in the presence of gaps and noise. In our method, we base our threshold on the spatial model rather than the temporal one. We introduce a minimum log likelihood tolerance θ_{min} , and filter out all assignments in each hypothesis with a lower log likelihood than θ_{min} . The filtered out assignments are given a uniform score of $\log(1/M)$.

In order to tune the statistical models, the method requires a short training sequence where the subject performs a full range of motion (RoM) of the markers. A good practice is to perform the training motion with special care taken to produce as few gaps as possible. The training data are manually labeled using the motion capture software or a simple MTT tracking algorithm. In our experience, this is straightforward and takes just a few minutes. Fig. 4 shows the training data for 5 markers on the fingertips. As can

³ <http://www.cyberglovesystems.com>.

Fig. 4. Left: Spatial distribution of a marker set with 5 markers on the finger tips. Right: fitted Gaussian Mixture Models with 3 mixture components per marker.

Fig. 5. Viterbi trellis displaying the evolution of the state machine having $N = 3$ hypothesis in time. χ_i^t denotes assignment hypotheses and s_i^t underlying temporal states. The red path shows the most probable path at the last frame. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

be seen, the spatial distribution of the markers contains several overlapping areas that account for ambiguities of the assignments. The training data are used to fit the GMMs for each marker using an Expectation Maximization (EM) algorithm. For the purpose of this study we found that 3 components in the GMM gave the best results for finger markers, and 1 component for face markers. We emphasize that our system models all markers individually and does not take inter-marker relations into account. Therefore training data does not need to be similar to test data as long as the individual marker distributions are representative.

3.2. Hypothesis selection

In this section, we describe how we select the best sequence of assignment hypotheses in time using the Viterbi algorithm and the Kalman filter. Given a sequence of latent states under a Markovian assumption, the Viterbi algorithm selects the most probable path in the trellis spanned by the discrete time sequence of states (see Fig. 5). The algorithm uses the probability at each time t of observing the data given each state (called the emission probability), and the probability of each state at time t to transition to each state in the next time step $t + 1$ (called transition probability). In our case, the emission probabilities are given by Eq. (1). To calculate the transition probabilities, we set up a total of $N \times M$ Kalman filters, one for each marker in each hypothesis. The motion of the markers is modeled as a dynamic system with velocity and acceleration, $\mathbf{s} = [\mathbf{x}, \dot{\mathbf{x}}, \ddot{\mathbf{x}}]$, and the system and observation noise parameters are manually tuned against the training data. Given two consecutive time frames $t - 1$ and t , each with N hypotheses, we find the transition probability $\mathbf{T}_{nm}$ for going from hypothesis χ_n^{t-1} to χ_m^t as follows. A prediction step for all Kalman filters in hypothesis χ_n^{t-1} is

performed, generating M predicted positions $\{\hat{\mathbf{x}}_{n,1}^t, \dots, \hat{\mathbf{x}}_{n,M}^t\}$. For each marker m_i we calculate the prediction residual (also called innovation) as the difference $\mathbf{r}_{im}^t$ between its predicted position and its observed position at frame t according to assignment hypothesis χ_m^t . The transition probability can be calculated from the Kalman filter as the sum of the log likelihood of all residuals

$$\mathbf{T}_{nm}^t = \sum_{i=1}^K \log(f(\mathbf{r}_{im}^t, 0, \mathbf{S}_{in}^{t-1})) \quad (3)$$

where $f(\mathbf{x}, \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate Gaussian probability given in Eq. (2), and $\mathbf{S}_{in}^{t-1}$ the residual covariance matrix for the current state of the Kalman filter for m_i in hypothesis χ_n . For more detailed information we refer to the section on Kalman filters in [25]. If a marker in the new hypothesis would be assigned as an occlusion, we extrapolate its trajectory by using the predicted state from the Kalman filter and omit the innovation update step. We limit the extrapolation to a short period of time (we found 10 frames to be a good limit). Longer gaps are reinitialized after the gap, in which case we do not calculate the temporal likelihood for the first two frames, after which the filter has stabilized to the new trajectory.

During online capture, each incoming frame consists of a set of unlabeled observations in 3D space. At the first frame, our algorithm calculates the N assignment hypotheses $\chi_1^0, \dots, \chi_N^0$ using the spatial model described in Section 3.1 and initializes the Kalman filters with the marker positions according to the assignment. The velocity and acceleration are set to zero. The following time frames proceed as follows:

- 1: Predict new positions for each of the markers in the previous frame's assignment hypotheses, using the corresponding Kalman filter.
- 2: Calculate N new assignment hypotheses for the current frame using the spatial model, taking no account of temporal information.
- 3: **for** each new hypothesis χ_m^t **do**
 - (a) Calculate the transitions probability matrix $\mathbf{T}$ according to Eq. (3).
 - (b) Determine the best transition to the new hypothesis using the Viterbi algorithm.
 - (c) Update the Kalman filters for each marker m_i according to the best path.
- 4: **end for**

3.3. Labeling Multiple Marker Sets

While the algorithm described above is adequate for situations where only one marker set is present in the volume, or when different marker sets are enough separated to be considered as independent, there are many practical situations where multiple marker sets interact closely, and it is hard to determine to which marker set an unlabeled marker belongs. In this section, we show how our algorithm naturally extends to handle multiple marker sets in close interaction by finding global hypotheses for all marker labels. To modify our algorithm for this purpose, we revisit the likelihood matrix in Eq. (1). The equation includes an affine transformation, $\mathbf{A}_i^t$, which transforms the unlabeled marker positions to the local space of the marker set. When multiple marker sets are present, each with an associated transformation $\mathbf{A}_i^t$, the global likelihood matrix, containing the likelihoods for all marker assignments, take the form

$$L_{ij}^t(\mathbf{y}_j^t) = \log \left(\sum_l (w_{li} f(\mathbf{A}_i^t \mathbf{y}_j^t, \boldsymbol{\mu}_{li}, \boldsymbol{\Sigma}_{li})) \right) \quad (4)$$

where $\mathbf{A}_i^t$ is the transform of the marker set to which marker i belongs. Note that the global likelihood matrix consist of the

Fig. 6. Experiment 1: Images from the data recording. Left: Ten markers placed on the fingertips and proximal joints. Right: Grasp motion.

vertically concatenated matrices of all marker sets and has the form $(M_{tot} \times K)$, where M_{tot} is the total number of markers. After constructing the global likelihood matrix, the hypotheses generation and hypotheses selection algorithms are applied as before.

The training data for tuning the marker set GMMs may be recorded in parallel or in separate takes depending on what is most practical. In our experiment with simultaneous face and finger capture, we perform the range of motion of both hands in one recording, and the face in another.

4. Evaluation

To evaluate our labeling algorithm, we preformed three experiments using independent data sets. In the first experiment, we evaluate the accuracy of the algorithm for labelling different configurations of finger markers. In the second experiment, we evaluate the algorithm in a large capture-volume and compare it to commercial systems. In the third experiment, we demonstrate how the method is used to simultaneously label finger and face markers in a full performance capture.

4.1. Experiment 1: Accuracy

In the first experiment, we evaluate the accuracy of our algorithm, and how generalizable it is to different marker sets. The data for this experiment was recorded in our motion capture lab, which is equipped with a NaturalPoint Optitrack⁴ system (Motive 1.10.0) with 16 Prime 41 cameras. The cameras have a resolution of 4 mega-pixels and were operated with a frame rate of 120 fps. The active capture volume is approximately 5 m × 5 m × 3 m.

The recordings consist of a 9000 frames long clip with a full range-of-motion of the finger joints, followed by a 9600 frames long clip with a series of grasping motions as well as a general exercising of the fingers. We used a marker set with 10 markers placed on the fingers according to Fig. 6a. After manually labeling the markers (approx. 20 min for the training data and 3 h for the test data), we generated 5 independent data sets (one for each of the marker layouts in Table 1) by filtering out some of the markers from the training and test data. Finally, we applied our algorithm to each data set, with parameters settings $N = 5$ and $\theta_{min} = -2$.

Table 1 shows the results of the labeling process. As can be seen in the table, the algorithm produced highly accurate results, with 99.67% to 99.98% correct labels and only a total of 80 erroneous labels for the most complex marker set with 10 markers.

To further our assessment, we manipulated the data to see how accurately our method can handle the initialization phase when (re)entering the volume, and how the initialization is affected by the number of visible markers. We hypothesized that scenarios where one or more markers is occluded during initialization would

Fig. 7. Experiment 2: Images from the 3 sets of recorded motion. (a) Marker placement. (b) Sign language. (c) Gestures. (d) Grasps.

be harder to label than scenarios where all markers were visible. To provide data for the test, we randomly selected 100 five seconds long snippets from the test data with 5 markers on the fingertips, and subsequently ran our labeling algorithm on each sequence four times, each time randomly removing more markers. Fig. 8 shows the average labeling errors. As can be seen, our algorithm can accurately handle initialization in general, with accuracy scores ranging from 0.23% to 4.10%. The worst score was obtained when only two markers were visible.

4.2. Experiment 2: Comparison to commercial systems

In the second experiment, we evaluate the method in a large capture volume (7 m × 12 m × 5 m), and compare it to commercial systems for marker labeling. The data for this experiment was recorded in a high-end professional motion capture studio providing services for film, commercials and AAA video games (see Fig. 2). The studio is equipped with a Motion Analysis system with 38 cameras. The cameras have a resolution of 4 mega-pixels and were operated at a frame rate of 120 fps. The recordings started with a 1.5 min range of motion followed by the three sets of test motion (see Fig. 7). The first set of test data consist of all letters in the Swedish Sign Language alphabet, the second of a variety of gestures such as pointing, thumbs up, stone-paper-scissors and boxing, and the third of grasping motion (using marbles, a spoon, a bottle, a paper and a mobile phone). The test sets were designed to provide a variety of challenging situations including uncommon finger poses, object-finger interactions and fast motion.

We prepared four versions of labeled data for each test set: (a) manually corrected (ground truth) labels; (b) using Vicon Blade (version 2.6.1) in offline mode (labeling skeleton set up according to Fig. 10); (c) using Motion Analysis Cortex (version 5.5.2) in online mode; and (d) using our method (with parameters $N = 5$, $\theta_{min} = -2$). Vicon Blade was run in offline mode as it does not support online labeling of imported data. The manual cleanup of the RoM data was performed by the studio technician and took a few minutes. The cleanup of the test data took about 4 h.

⁴ <http://www.optitrack.com>.

Table 1

Experiment 1: Labeling results for different marker configurations for the 9629 frames of test data. The number of instances is given by (number of frames) $\times$ (number of markers). We separately account for correct labels (markers labeled with the right label or correctly marked as a gap), erroneous labels (markers labeled with the wrong label), false markers (occlusions labeled as markers) and false occlusions (markers labeled as occlusions), as well as the number of gaps and mean gap and segment length.

Marker set
#instances	19,258	28,887	48,145	57,774	96,290
#correct labels	19,253 (99.97 %)	28,882 (99.98%)	47,986 (99.67%)	57,639 (99.77%)	96,049 (99.75 %)
#erroneous labels	0 (0.00%)	0 (0.00%)	99 (0.21%)	78 (0.14%)	80 (0.08 %)
#false markers	0 (0.00%)	0 (0.00%)	20 (0.04%)	16 (0.03%)	66 (0.07 %)
#false occlusions	5 (0.03%)	5 (0.02%)	40 (0.08%)	41 (0.07 %)	95 (0.10 %)
#gaps	66	63	180	192	552
mean gap length	11 frames	10 frames	11 frames	11 frames	9 frames
mean segment length	274 frames	426 frames	248 frames	281 frames	161 frames

Table 2

Experiment2: Comparison of the accuracy of labeling the three sets of test data (Signs, Gestures and Grasps) with Vicron Blade v2.5.6, Motion Analysis v5.5.2 and our method. The data contain a total of 11,383 frames (56,915 instances) for the Signs data, 10,691 frames (53,455 instances) for the Gesture data and 13,067 frames (65,335 instances) for the Grasp data.

Labeling system	Test set	#correct labels	#erroneous labels	#false markers	#false gaps
Blade	Signs	51,878 (91.15%)	4977 (8.7%)	21 (0.04%)	39 (0.07%)
	Gestures	53,329 (99.76%)	4 (0.01%)	0 (0.00%)	122 (0.23%)
	Grasps	49,825 (76.26%)	15,014 (22.98%)	232 (0.35%)	264 (0.40%)
Cortex	Signs	56,624 (99.49%)	226 (0.40%)	17 (0.03%)	48 (0.08%)
	Gestures	42,533 (79.56%)	10,085 (18.87%)	387 (0.72%)	450 (0.84%)
	Grasps	52,074 (79.70%)	12,638 (19.34%)	210 (0.32%)	413 (0.63%)
Our	Signs	55,490 (97.50%)	853 (1.50%)	269 (0.47%)	303 (0.53%)
	Gestures	53,142 (99.41%)	61 (0.11%)	84 (0.16%)	168 (0.31%)
	Grasps	64,800 (99.18%)	39 (0.06%)	77 (0.11%)	419 (0.64%)

Fig. 8. Experiment1: Labeling errors for different number of visible markers, averaged over 100 randomly selected 5 s sequences.

We then compared the manually corrected ground truth labels with the output from the different labeling algorithms. Fig. 9 shows a color-coded sequence of all the labels. Correctly labeled markers are coded as green, correctly labeled occlusions as blue, erroneous labels as red, false markers (occlusions labeled as markers) as cyan and false occlusions (markers labeled as occlusions) as yellow. As can be seen in Table 2 our method performed best on average and was especially stable for the Gesture data set, which had the highest amount of fragmentation with 255 gaps (see Table 3).

Table 3

Experiment2: Number of gaps and mean gap and segment length, where a segment is a trajectory fragment surrounded by two gaps.

Test Set	#gaps	Mean gap length	Mean segment length
Signs	38	14 frames	1310 frames
Gestures	255	5 frames	198 frames
Grasps	89	6 frames	687 frames

4.3. Experiment 3: Performance capture

In the third experiment, we evaluate our algorithm when used to simultaneously label face and finger markers in a full performance capture. We also demonstrate how our method is used together with data-driven marker reconstruction to provide high quality data from sparse marker sets. For this experiment, we used a subset of the data from a previous study on expressive artificial agents captured in our motion capture lab [26]. The data consist of an 8.4 min (60,315 frames) long motion capture clip of an actor giving instructions to an interlocutor, while varying the displayed level of engagement from very un-engaged to very engaged, as well as two shorter clips of RoM data (one for the hands/fingers and one for the face). The actor wore 43 markers placed on the body, 5 markers on the fingertips of each hand, and 19 markers on the face (Fig. 11a). In the RoM clips, additional markers were placed on the fingers and face, providing a total of 10 markers on the fingers of each hand and 36 markers on the face (Fig. 11 b). The labeling procedure was performed in two steps. First, the body markers (excluding fingers and face) were labeled using the kine-

Fig. 9. Experiment2: Visualization of labeling results. Green - correct labels, Blue - correct occlusions, Red - erroneous labels, Cyan - false markers, Yellow - false occlusions. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 4

Experiment3: Results from simultaneous labeling of face and finger markers. The data set contains 60,315 frames and 29 unlabeled markers (1,749,135 instances).

Marker set	#correct labels	#erroneous labels	#false markers	#false gaps
Face	1,145,679 (99.97%)	5 (0.0004%)	0 (0.00%)	301 (0.03%)
Left fingers	301,214 (99.88%)	34 (0.01%)	41 (0.01%)	286 (0.09%)
Right fingers	301,317 (99.91%)	12 (0.00%)	59 (0.02%)	187 (0.06%)

Fig. 10. Experiment2: Labeling skeleton for setup and calibration in Vicon Blade.

Fig. 11. Experiment3: (a) Marker set used for the test data. (b) Marker set with additional markers used for the range-of-motion.

matic labeler of the Natural Point Motive software. Thereafter, the remaining unlabeled markers were fed into our system together with the calculated transforms of the head and hands (with parameters $N = 15$, $\theta_{min} = -2$). As can be seen in Table 4 the output resulted in 99.88% to 99.97% correct labels.

After labeling was completed, we reconstructed the additional markers in the RoM take using the method given in [19]. This method was originally developed for data-driven mesh deformation using a small set of control points, but, as markers have similar properties, it also applies to missing marker reconstruction. The algorithm uses Kernel Canonical Correlation Analysis (kCCA), trained on pairs of example mappings between two multivariate spaces, to produce new estimates given unseen data. For the finger data, we reconstructed each of the markers on the proximal phalanges individually using the fingertip marker on the same finger as input data. For the face data, we reconstructed all the extra markers in the RoM set except the markers on the lower eyelid, which proved to have too much uncorrelated motion to yield a satisfying result. Here we used the complete sparse marker set as input data to the kCCA algorithm. We then solved the skeleton and facial motion using commercial software (Autodesk Motion Builder and IKinema Action for the body and fingers, Softimage Face Robot for the face). Images from the performance capture and corresponding character animation are shown in Fig. 12.

To estimate errors for the reconstruction process, we performed a series of test on the RoM data. Fig. 13 shows displacement errors for reconstructed markers after 10-fold cross validation, where subsequent tenths of the data was held out and reconstructed using the rest of the data for training. As can be seen in the figure, the mean errors ranged between 0.4 mm and 1.9 mm for the reconstructed face markers, and between 4.7 mm and 6.4 mm for the finger markers.

5. Discussion

The results of the first experiment show that our method obtains high accuracy for a wide variety of marker configurations for finger capture. Unsurprisingly, the marker sets with 2 and 3 markers per hand, having best separation and fewest gaps, were least challenging for our algorithm, which generated almost perfect results without any erroneous labels and only a few false gaps. More interestingly, the results for the challenging marker sets with 5 to 10 markers were almost as good, with accuracy scores ranging from 99.7% to 99.8%. Although our method does not utilize an underlying skeleton model, the results are well on par with the

By inspecting the results of the second experiment comparing different labeling algorithms, the majority of the errors from our method occurred during the first set of sign recordings, and were caused by finger poses outside the range-of-motion in the training data. For example, the sign for the letter ‘X’ has crossed index and middle finger, and the sign for the letter ‘R’ has extended middle finger, while the rest of the fingers are clenched in a fist. The shorter stretches of swapped markers in Fig. 9, lower left, arise from these signs. When the system has resided in the unfamiliar pose for some time, the Viterbi algorithm favours swapping to a more probable assignment over maintaining temporal smoothness. This introduces a sudden discontinuity in the marker trajectories. The swaps from the Blade and Cortex labelers, however, occur after gaps and the markers remain swapped until the next gap occurs. Consequently, as can be seen in Fig. 9, these systems yield much longer periods of swaps, which implies that our system is particularly better for real-time applications. A possible way to improve the results from our system would be to add more poses to the training set. Another possibility is to introduce a manually tuneable parameter to increase the weight of temporal smoothness over the likelihood from the spatial model. As a final remark to experiment 2, it can be noted that one of the systems (Vicon Blade) in the comparison is run in offline mode, which theoretically gives it an advantage over the other systems. Yet in our tests, it is outperformed by our method.

As seen in experiment 3, our algorithm also obtains high accuracy for labelling finger and face markers in a full performance capture. One problem we experienced in this experiment came from the procedure of labeling the body markers in a separate step before-hand. In this process, one of the finger markers were occasionally mislabeled as belonging to the body. These errors were manually corrected prior to feeding the unlabeled markers to our system. A limitation of our method is that it splits the labeling procedure into two stages, one for body/rigid bodies and one for non-rigid structures. Ideally, to prevent erroneous labels in the first

stage, our method should be incorporated in the full body/rigid body algorithms.

6. Conclusions and future work

We have presented a system for robust online labeling of passive markers attached to non-rigid structures such as fingers and faces. The system was evaluated on a variety of marker sets captured in medium to large volumes and was shown to provide high accuracy results for all cases. In a comparison with commercial systems, our method was shown to be more robust and produce better results on average. The method is especially beneficial for reduced marker sets and data-driven methods for hand and face solving, but it also shows accurate results for larger marker sets with 10 markers on the finger-tips and proximal joints.

In future work we aim to optimize the code to improve performance and to create preset databases of training data for different hands anatomies. We will also investigate ways to incorporate data-driven methods for automatic gap-filling in our labeling process. In our current system, gap-filling is performed after labelling is complete.

Acknowledgments

We wish to thank Anton Söderhäll and Samuel Tyskling at Imagination Studios for providing motion capture services used in this study as well as valuable feedback and discussions on the state-of-the-art in the industry. We also thank Ludovic Hoyet for discussions and for labeling the second data set with Vicon Blade.

This work was funded by KTH/SRA ICT The Next Generation and Science foundation Ireland PI grant #S.F. 10/IN.1/13003.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at [10.1016/j.cag.2017.10.001](https://doi.org/10.1016/j.cag.2017.10.001).

References

- [1] Wheatland N, Wang Y, Song H, Neff M, Zordan V, Jörg S. State of the art in hand and finger modeling and animation. *Comput Graph Forum* 2015;34(2):735–60.
- [2] Wheatland N, Jörg S, Zordan V. Automatic hand-over animation using principle component analysis. In: *Proceedings of motion on games*. ACM; 2013. p. 197–202.
- [3] Schröder M, Maycock J, Botsch M. Reduced marker layouts for optical motion capture of hands. In: *Proceedings of the 8th ACM SIGGRAPH conference on motion in games*. ACM; 2015. p. 7–16.
- [4] Mousas C, Newbury P, Anagnostopoulos C-N. Efficient hand-over motion reconstruction. In: *Proceedings of the 22nd International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision*; 2014. p. 111–20.
- [5] Alexanderson S, O'Sullivan C, Beskow J. Robust online motion capture labeling of finger markers. In: *Proceedings of the 9th international conference on motion in games*. ACM; 2016. p. 7–13.
- [6] Kitagawa M, Windsor B. *MoCap for artists: workflow and techniques for motion capture*. CRC Press; 2012.
- [7] Aristidou A, Lasenby J. Motion capture with constrained inverse kinematics for real-time hand tracking. In: *Proceedings of the 4th International Symposium on communications, control and signal processing (ISCCSP)*, 2010. IEEE; 2010. p. 1–5.
- [8] Gibet S, Courty N, Duarte K, Naour TL. The signcom system for data-driven animation of interactive virtual signers: methodology and evaluation. *ACM Trans. Interact Intell. Syst. (TiiS)* 2011;1(1):6.
- [9] Maycock J, Rohlig T, Schroder M, Botsch M, Ritter H. Fully automatic optical motion tracking using an inverse kinematics approach. In: *Proceedings of the 15th international conference on humanoid robots (Humanoids)*, 2015 IEEE-RAS. IEEE; 2015. p. 461–6.
- [10] Bar-Shalom Y. *Tracking and data association*. Academic Press Professional, Inc.; 1987.
- [11] Reid DB. An algorithm for tracking multiple targets. *IEEE Trans. Autom. Control*, 1979;24(6):843–54.
- [12] Gennari G, Chiuso A, Cuzzolin F, Frezza R. Integration of shape constraints in data association filters. In: *Proceedings of the 43rd IEEE conference on decision and control*, 2004. CDC, vol. 3. IEEE; 2004. p. 2668–73.
- [13] Yu Q, Li Q, Deng Z. Online motion capture marker labeling for multiple interacting articulated targets. *Comput Graph Forum* 2007;26(3):477–83.
- [14] Ringer M, Lasenby J. Multiple hypothesis tracking for automatic optical motion capture. In: *Proceedings of the european conference on computer vision*. Springer; 2002. p. 524–36.
- [15] Meyer J, Kuderer M, Muller J, Burgard W. Online marker labeling for fully automatic skeleton tracking in optical motion capture. In: *Proceedings of the IEEE international conference on robotics and automation (ICRA)*. IEEE; 2014. p. 5652–7.
- [16] Schubert T, Gkogkidis A, Ball T, Burgard W. Automatic initialization for skeleton tracking in optical motion capture. In: *Proceedings of the 2015 IEEE international conference on robotics and automation (ICRA)*. IEEE; 2015. p. 734–9.
- [17] Akhter I, Simon T, Khan S, Matthews I, Sheikh Y. Bilinear spatiotemporal basis models. *ACM Trans. Graph. (TOG)* 2012;31(2):17.
- [18] Hoyet L, Ryall K, McDonnell R, O'Sullivan C. Sleight of hand: perception of finger motion from reduced marker sets. In: *Proceedings of the ACM SIGGRAPH symposium on interactive 3D graphics and games*. ACM; 2012. p. 79–86.
- [19] Feng W-W, Kim B-U, Yu Y. Real-time data driven deformation using Kernel canonical correlation analysis. In: *ACM Trans. Graph.*, vol. 27. ACM; 2008. p. 91.
- [20] Alexanderson S, Beskow J. Towards fully automated motion capture of sign-s-development and evaluation of a key word signing avatar. *ACM Trans Access Comput (TACCESS)* 2015;7(2):7.
- [21] Kalman RE. A new approach to linear filtering and prediction problems. *J Basic Eng* 1960;82(1):35–45.
- [22] Forney GD. The viterbi algorithm. *Proc IEEE* 1973;61(3):268–78.
- [23] Taha HA. *Operations research: an introduction* (8th Edition). Upper Saddle River, NJ, USA: Prentice-Hall, Inc.; 2006. p. 0131889230.
- [24] Murty KG. Letter to the editor an algorithm for ranking all the assignments in order of increasing cost. *Oper Res* 1968;16(3):682–7.
- [25] Murphy KP. *Machine learning: a probabilistic perspective*. MIT press; 2012.
- [26] Alexanderson S, O'Sullivan C, Neff M, Beskow J. Mimebot - investigating the expressibility of non-verbal communication across agent embodiments. *ACM Trans Appl Percept (TAP)* 2017;14(4):24.