 doi:10.20944/preprints202203.0249.v1

Article

Detection and Classification of Artifact Distortions in Optical Motion Capture Sequences

Przemysław Skurowski [1]* and Magdalena Pawlyta [1,2]

[1] Department of Graphics, Computer Vision and Digital Systems, Faculty of Automatic Control, Electronics and Computer Science, Silesian University of Technology, Akademicka 16, 44-100 Gliwice, Poland; Przemyslaw.Skurowski@polsl.pl (P.S.); Magdalena.Pawlyta@polsl.pl (M.P.)

[2] Polish-Japanese Academy of Information Technology, Koszykowa 86, 02-008 Warsaw, Poland; mpawlyta@pjwstk.edu.pl

* Correspondence: przemyslaw.skurowski@polsl.pl; Tel.: +48-32-2372151

Abstract: Optical motion capture systems are prone to the errors connected with markers recognition – occlusion, leaving the scene or mislabelling – all these errors are then corrected in the software, but still, the process is not perfect, resulting in artifact distortions. In the article, we examine four existing types of artifacts, then propose the method for detection and classification of the distortions. The algorithm is based on the derivative analysis, low-pass filtering, mathematical morphology and loose predictor. The tests involved multiple simulations using synthetically distorted sequences, comparison of performance to the human operators on real life data and applicability analysis for the distortion removal.

Keywords: motion capture; artifact classification; artifact detection; reconstruction;

1. Introduction

Motion capture (mocap) systems [1,2] play important role in modern computer graphics, where they are applied in gaming and movie FX as a mean for generation of realistically looking animation of characters. Prominent applications of mocap are also in biomechanics and medical sciences [3]. Up today, the most reliable technology is marker based optical mocap (OMC) – it is sometimes called 'gold standard', as it outperforms the other mocap technologies. It utilizes visual tracking of active or retro reflective passive markers. Trajectories of these markers are then used for animation of an associated skeleton, which is used as a key model for animation of a human-like or animal characters.

The process of acquisition of marker locations is error prone. Distortions, occurring in the mocap sequences, can be simply divided into two classes - random noise and algorithmically introduced artifact distortions. The random noise is a consequence of stochastic processes resulting in different kinds of distortions in mocap sequence. It was studied in numerous works [4–7]. Among the types of noise the most prominent [8] is white noise, which can be efficiently filtered out [9], or 'smoothed' – there were numerous methods proposed [10] utilizing low pass filtering, interpolating methods or moving averages. The artifact distortions are introduced by reconstruction algorithms present in mocap pipeline, and they can be considered as momentary systematic error. These distortions introduce trajectory modifications of a different appearance and of a larger amplitude. These trajectory mis-shapes are poorly filtered out by simple noise removing algorithms. The common practice in mocap studios is manual editing of the data by operators who visually examine and correct the trajectories.

In the article, we demonstrate a marker-wise method for detection and classification of these systematic errors. The key motivation for development of the classifier of distortions is the fact, that for each of different distortion classes, we can select appropriate method of suppressing - e.g. for the rectangular distortion, which is result of mistaken marker labeling, it is enough to find its counterpart marker and to swap erroneous parts of trajectories.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

The proposed approach is skeleton-free, and therefore it is able to adapt to virtually any vertebrate subject. There are two basic assumptions - rigid body model and correlation of marker trajectories. First, a rigid body model is assumed for the obtaining functional body mesh (structured point cloud) [11], which we use to represent the subject's body hierarchy. The next assumption stems from the former, it is the fact, that movement of markers are highly correlated and predictable when they are placed on the common body parts (e.g. limbs) – we employ a deviation of a trajectory from the prediction as a criterion for classification.

The article is organized as follows: in chapter 2 we disclose the background for the article – mocap pipeline with sources of distortions and former works on the distortions in optical mocap systems; chapter 3 describes the proposed method with its rationales and design considerations. In the chapter 4 we test the method for its performance and discuss the results. Chapter 5 summarizes the article.

2. Background

2.1. Sources and types of distortions

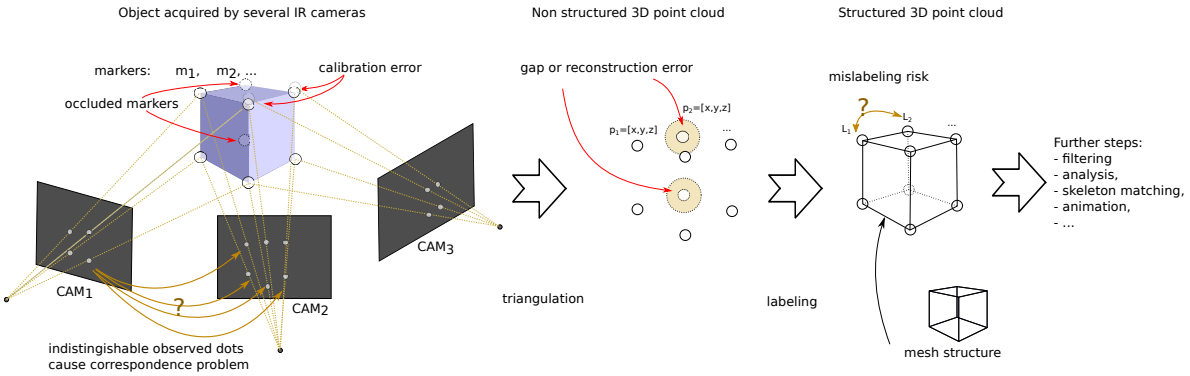

Figure 1. Processing in the early stages of the motion capture pipeline with sources of distortions (red) and problems to solve (yellow)

In the optical mocap marker tracking is obtained by image registration of marker position by multiple IR cameras. Multi view observation allow for reconstruction of marker trajectories (3D positions over time) through the triangulation of 2D position recordings, which are registered by multiple cameras. The process is error prone and various sources of the distortions can be identified, as it is depicted in Fig. 1. Besides conventional stochastic noise, two main sources of errors in marker registration are gaps and erroneous marker matching. Gaps occur when the marker disappears from cameras' view, it happens or because of locating body part outside of scene (camera range), either covering markers with another body part (occlusion). In such cases reconstruction algorithms can be source of errors. Marker matching happens twice in the mocap pipeline. First, prior to the triangulation, it is necessary to perform marker matching in multiple 2D views of single frame to identify corresponding 2D locations of markers. Another marker matching procedure is labeling (naming) – performed among the different frames where it is necessary to identify corresponding successive positions of 3D markers in sequence of frames. All these procedures can result in one trivial and four regular types of distortions, which can be observed in current mocap systems. These are:

1. simple gap - appearing when reconstruction algorithms gave up, type of least concern as a trivial case,
2. single peak - caused by transient erroneous marker matching, simple to detect,
3. heavy noise of a much larger amplitude than ordinary noise introduced by frequent erroneous marker matching,
4. rectangular distortion - forward followed by backward step caused by mismatching of 3D positions of markers (some part of 3D trajectory is assigned to another marker), or due to the erroneous marker reconstruction based on rigid body model,

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

5. slow value change - two potential sources – accumulated reconstruction error in successive frames (e.g. when there is deformation of a body which is failure of commonly assumed a rigid body model), or a result of low-pass filtering of peaks.

All above classes are depicted in Fig. 2. They can be roughly divided into two basic classes - sudden (2-4) and slow (5) changes to the trajectory.

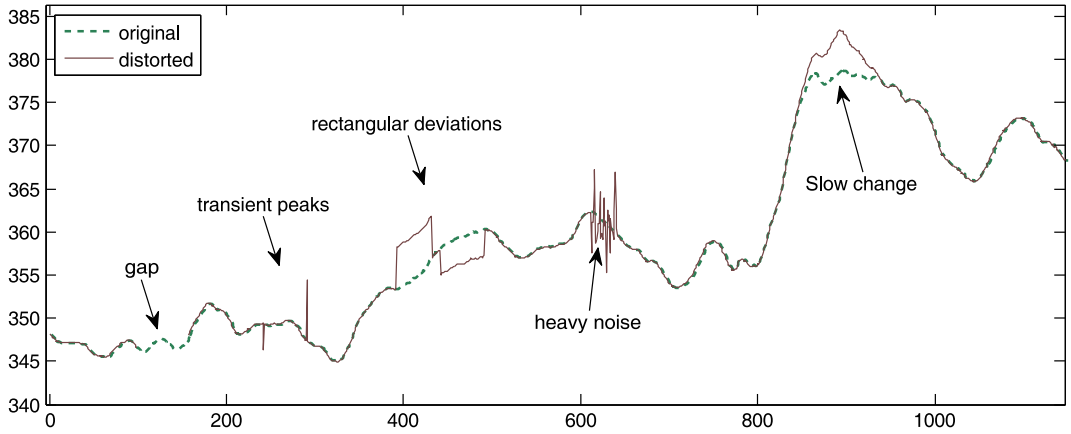

Figure 2. Identified types of distortions inpainted into exemplary data - first coordinate of first marker (head) of IM subject

2.2. Previous works

To the authors' knowledge, this work is pioneering in the identification of artifact distortions in mocap sequences. Obviously, the knowledge of reconstruction imperfections was present before in the former works. Therefore, there are quite a lot of works in motion capture area related to the occlusion gap filling problem. They involve various methods for signal reconstruction, when the marker is lost in recorded sequence [12–17]. Approaches, which were are proposed for that purpose include (among the others): rigid body model, fusion of weak regressors, inverse kinematics, neural networks. They differ in the assumptions, performance, and complexity, though the main aspect which makes certain method to be suitable or not is the length of the gap – simple signal based methods (e.g. interpolation) works well for short errors whereas complex, model based, methods suit better for the long gaps.

The other approach is taking certain stages of the pipeline into consideration. Their authors focus on partial problems in the motion capture pipeline, and they perfect these individual steps – improving system configuration [18] (e.g. number and layout of cameras) and calibration [19] and labelling [20–23].

3. The proposal

3.1. Premises – correlation of trajectory coordinates

The correlation of locations and gradients of markers is a key rationale that that allowed us to propose a method to classify all the types of distortions. Since the variables in Mocap sequences can be strongly (positive and negative) correlated within the groups, thus artifacts introduced by reconstruction algorithms should differ significantly enough to distinguish them on the basis of the proper trajectories of neighboring markers.

In Fig. 3a we present correlation coefficient (CC) in a form of a distance matrix. It demonstrates structural dependencies in the correlation between the marker positions. One can easily observe the clusters formed by the body parts – hands, torso, legs and so on. The correlation is high within individual body parts (both positive and negative), on the other hand the correlates between body

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

parts would depend on the registered motion – in case of natural walking the hands position would be counter-correlated, whereas in butterfly swimming motion the hands position would be correlated.

The time aspect of correlation is depicted in Fig. 3b., it presents correlation (and autocorrelation) function for several marker pairs. It clearly illustrates the correlation between successive marker locations and furthermore between locations of the markers located in the common body segments and connected body segments (e.g. head and neck).

To conclude the reasoning, we could suppose that we can reliably identify the outstanding markers on the basis of the markers correlated within groups – from the common body segment or parent body part.

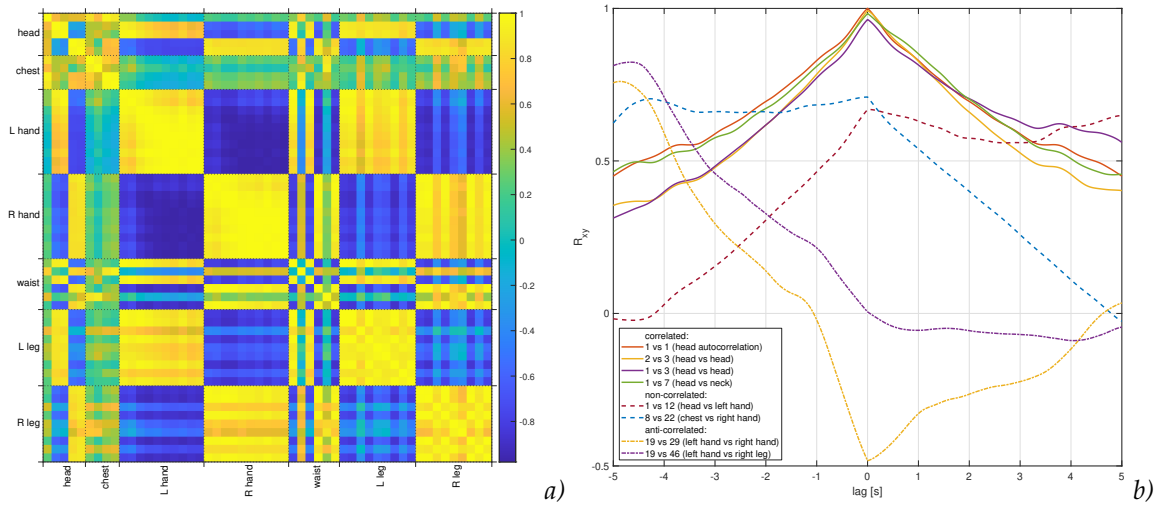

Figure 3. Correlation between X position of markers in exemplary sequence (fast walking HJ subject): a) the whole sequence, all 53 markers; b) inter-marker correlation function for selected correlated and non-correlated markers

3.2. The method overview

The key idea of the algorithm is to use model (prediction) results as a verification criterion for the data. If the data fall too far from the prediction results then they are rationale to consider it as a distortion, that can be assigned into one of the distortion classes using pattern recognition methods. Each class of distortions is identified as a separate stage and it is cleaned form the signal by interpolation, then it is passed to next stage of detection. From the simples distorion (single peaks) to the most difficult (slow changes). Overall proces is depicted in Fig. 4.

The proposal is feasible thanks to the data correlation in MoCap sequences, which makes the prediction feasible, and which allows for reliable estimation of the actual position of a marker.

The methods for error identification and classification depend on the type of distortion. Sudden changes to the trajectory are identified on the basis of the differential of the signal and a low pass filter (as a predicting model) with stats based thresholding and mathematical morphology. It allows to distinguish between types of sudden changes. Slow changes detection is based on the hysteresis thresholding of residuals with backward re-growing of identified segments.

Three predictive models are employed in our pipeline. In sudden changes detection, which is the simpler case, we employ Savitzky-Golay and moving median filters to identify legitimate changes in the signal. For the detection of slow changes, we employed neighbor based predictors – initially we assumed polynomial predictor based on least squares method, which we gave up in favor of feed forward neural network (FFNN), yet we decided to include it into the description, as it depicts the development process.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

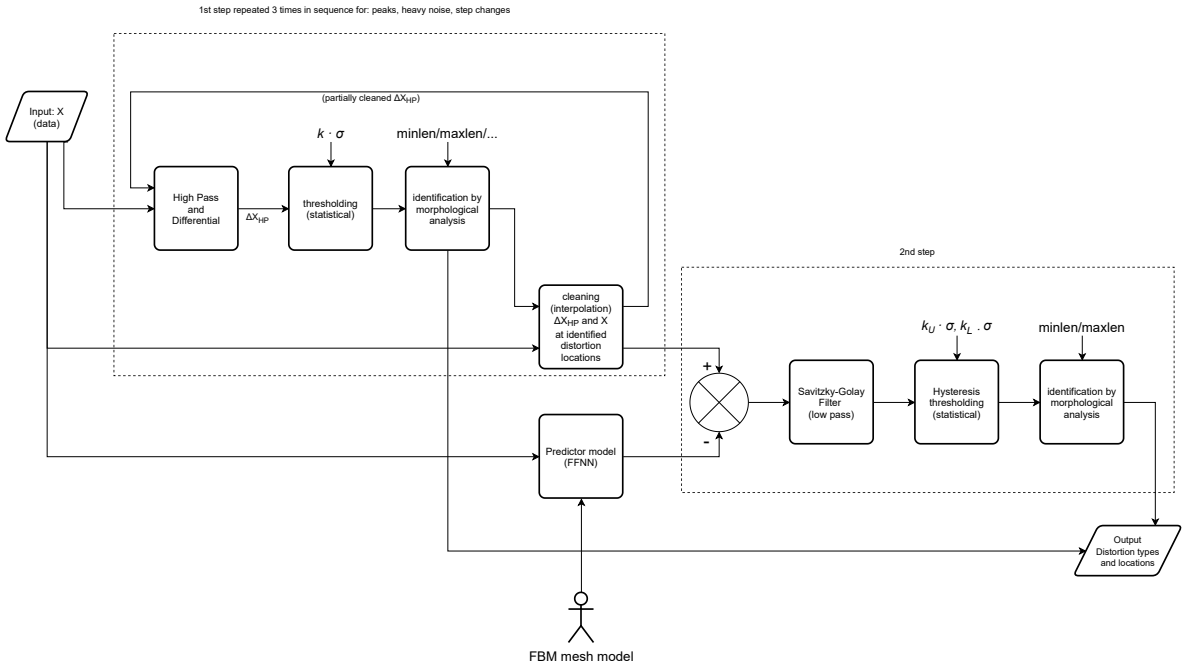

Figure 4. Conceptual schematic of proposed algorithm

3.3. Regressive models

The efficiency of overall approach would depend on the quality of model predictors and their ability to approximate the real location of a given marker (a location where it should be), on the basis of own or neighboring markers current, past or future locations. In general case the regression model (predictor) [24] has a form of a function:

$$\hat{Y}_i = f(X_i, \beta_i) + R_i, \tag{1}$$

which in the linear [25] case yields:

$$\hat{Y}_i = X_i \beta_i + R_i, \tag{2}$$

where for N observations of M regressor variables, $\hat{Y}_i$ is N-element long column of predicted values of i-th variable, X_i is the N-by-M design matrix for the model, β_i is N-element column vector of coefficients, R_i is error (residual). Model coefficients are estimated on the basis of X_i and Y_i a column vector of goal values with least squares method (LSM) denoted as:

$$\beta_i = (X_i' X_i)^{-1} X_i' Y_i. \tag{3}$$

The residual is remaining value, which is non predicted and non correlated part of the signal, that is given simply as:

$$R_i = Y_i - \hat{Y}_i. \tag{4}$$

The part which will be further analyzed is residual. It's probability in linear case follows Laplace (double exponential) distribution:

$$f(x|\mu, b) = \frac{1}{2b} e^{\left(-\frac{|x-\mu|}{b}\right)}, \tag{5}$$

where: μ is mean value equal to zero in our case, b is a dispersion parameter calculated on the standard deviation as $b = \sigma/\sqrt{2}$. Therefore, the standard deviation of residual (denoted as σ_R) can be employed for evaluating quality of prediction.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

3.3.1. Savitzky-Golay filter

Savitzky-Golay filter [26], is a smoothing filter, which is based conceptually on polynomial fitting in the least squares sense (there are efficient convolution based implementations). Its output is a value of polynomial function fit locally to the data. The coefficients (c^l) of L-th order are fit to the data within the sliding window of a size M centered around $x(i)$; filter output is polynomial value for the midpoint. In basic variant the filter is low-pass, but high-pass can be obtained by simple difference between signal and its smoothed variant:

$$p_{LP}(i) = \sum_{l=0}^{L} c_l \cdot x^l(i) \tag{6}$$

$$p_{HP}(i) = c(i) - p_{LP}(i) \tag{7}$$

Its least squares design matrix can be simply noted as:

$$X_i = \begin{pmatrix} 1 & x(i-M) & x^2(i-M) & \cdots & x^L(i-M) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x(i+M) & x^2(i+M) & \cdots & x^L(i+M) \end{pmatrix},$$

3.3.2. Neighbor based linear least squares loose model

Slow changes detection requires the predictor to be able to avoid following slowly accumulating changes in the signal, hence we decided to employ *loose* (weak) prediction, which does not rely on the own momentary positions of the marker, but uses only past and current positions of sibling and parent markers. Initially we employed polynomial model, which obtaining with ordinary least squares (LS) was conveniently planned using Vandermonde matrix ${}_iX$ (with some caution as it could be ill conditioned with growing polynomial order) as:

$$X_i = \begin{pmatrix} 1 & x_j(1) & x_j^2(1) & \cdots & x_j^L(1) & x_k(1) & x_k^2(1) & \cdots \\ 1 & x_j(2) & x_j^2(2) & \cdots & x_j^L(2) & x_k(2) & x_k^2(2) & \cdots \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \vdots & \cdots \\ 1 & x_j(N) & x_j^2(N) & \cdots & x_j^L(N) & x_k(N) & x_k^2(N) & \cdots \end{pmatrix},$$

where: $x(n)$ are successive 1..N values of a single regressor variable, L is a polynomial order, $i, j, k, \cdots$ are variable indices, such that $i \neq j, k, \cdots$.

Considering the predictor, there appears term order twice meaning the context size – number of former values taken into prediction, and polynomial order. Hence, to avoid confusion in the paper, we use following notation for predictors and residual:

$$P_k^L(i,n), \quad R_k^L(i,n) \tag{8}$$

where: L - means polynomial order used in X, k - number of past values of regressor variables used to construct X, n - number (time) of frame in sequence, i is a number of predicted variable.

The selection of the proper markers to formulate predictor for each marker according to Eq. (3) is based on the body hierarchical structure. For that purpose we used a body of a structure depicted in Fig. 5a which is a functional body mesh (FBM) [11] for an average human subject inferred for the typical Vicon marker setup. The FBM hierarchy with corresponding skeleton is shown in in Fig. 5b. The FBM represents a kinematic structure as groups markers located on the structure parts of a body and a hierarchy as a tree of these groups. The structure obtaining step has to be performed for each class of subjects or different set of markers separately. The design matrix X consisted of coordinates of parent and siblings in current and k former frames (but excluding former coordinates of the considered

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

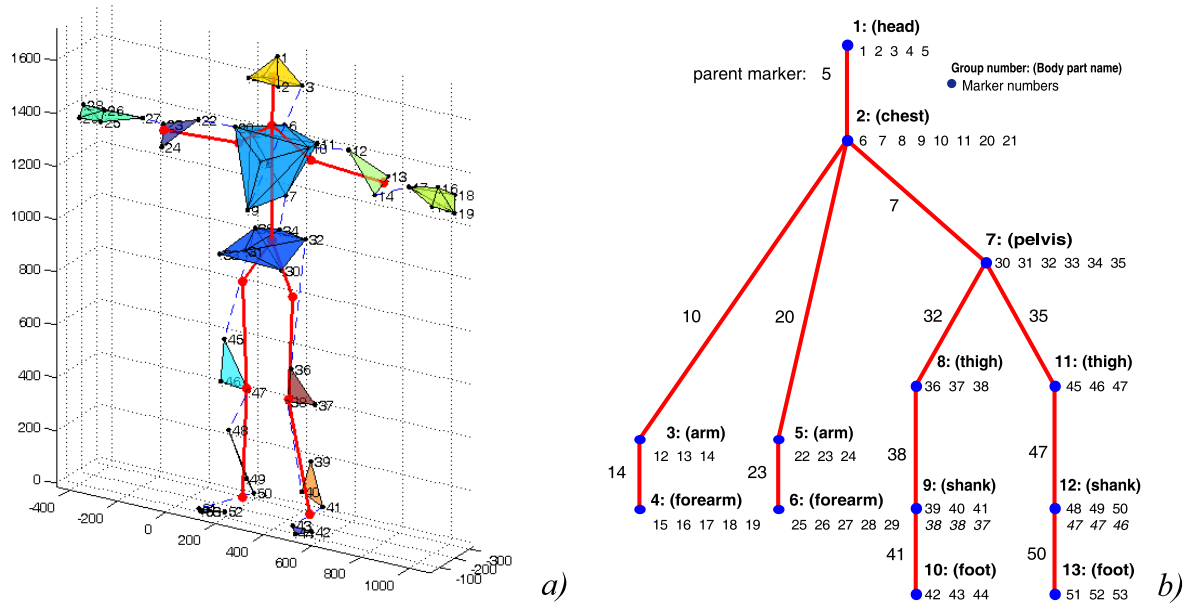

Figure 5. Outline of the body model (a), and corresponding parts hierarchy annotated with parents and siblings (b).

marker). A parent cluster is represented by a single marker - the one which is closest in terms of gradients coherence and distances constancy.

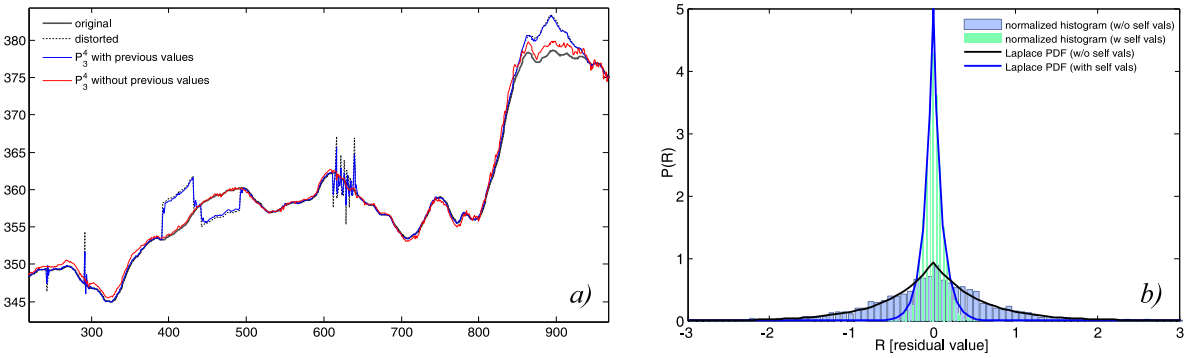

Figure 6. Performance of two predictor models of P_3^4 (with and without former values of predicted variable): a) first dimension of first marker (with artificial distortions); b) residual histograms and corresponding Laplace PDFs for R_3^4 (for explanation of model construction and parameters see Eq. 8).

Our demands towards the predictor are slightly specific. First, obvious requirement is that it to be as precise as possible, although, the other and contradictory requirement which is pivotal for us that it would not follow momentary changes which are induced by artifacts. Such requirements made us to choose special approach to formulate X matrix as predictors efficiency of predictor depends on the data that is used. We had to neglect past values of a considered marker - it is due to the fact that there is largest correlation between current and past location of a marker and therefore it ensures the accuracy (Fig. 6b), unfortunately, in case of a distortion it would make the predictor to follow the artifact deviation see Fig. 6a. Next, we had to choose predictor parameters - usually the higher degree of polynomial and context size are used the higher precision is obtained although due to ill conditioning of X it can reach higher error with growing polynomial order. Moreover, too large increasing of the context, would also not improve the the predictor accuracy. The predictor parameters were tuned with numerical testing with preliminary data. We decided to set up parameters to $k = 3$ and $L = 4$ as they appeared during the preliminary model tuning (Fig. 7) to be reasonable trade off between predictor accuracy and computation complexity.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

Summarizing each row vector in X is long and is assembled of certain parts as given below:

$$X(n) = \left[\begin{array}{c} \overbrace{1,\ \overbrace{x_p(n), y_p(n), z_p(n), x_p(n-1), \cdots, z_p(n-k)}^{\text{current value and } k \text{ former of parent marker } (p)},}^{\text{current value and } k \text{ former of first..last siblings}} \\ \underbrace{x_{s1}(n), y_{s1}(n), z_{s1}(n), x_{s1}(n-1), \cdots, z_{slast}(n-k),}_{\text{current value and } k \text{ former of first..last siblings raised to } L\text{th power}} \\ \vdots \\ x^L_{s1}(n), y^L_{s1}(n), z^L_{s1}(n), x^L_{s1}(n-1), \cdots, z^L_{slast}(n-k) \end{array} \right]. \quad (9)$$

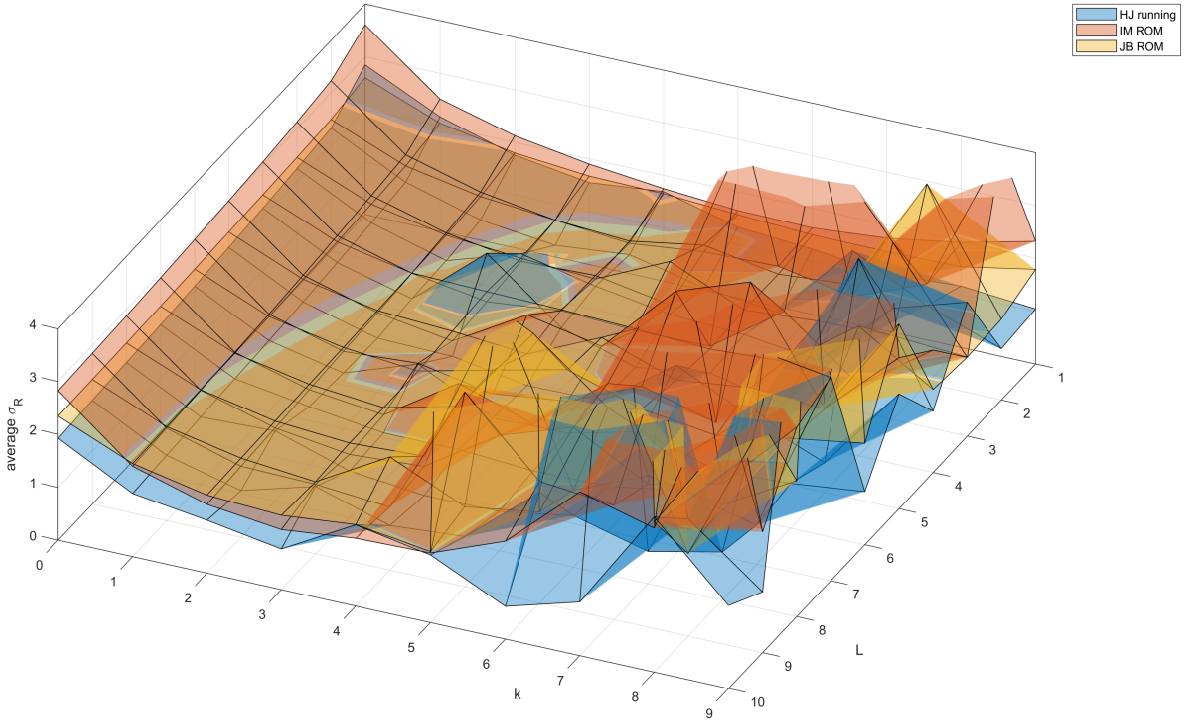

Figure 7. Predictor parameter tuning with preliminary data for three subjects, quality as standard deviation averaged over all markers; the tuned parameters parameters: a) polynomial order, b) context size (lags).

We studied the polynomial models thoroughly scanning parameter space (see Fig. 7), to obtain accuracy allowing for well identification of slow deviations, alas, the residual was noisy and the deviations were visible, but cluttered making their automatic identification (see p. 3.5) work poorly. Therefore we reviewed a series of various regression techniques – SVM, ANFIS-fuzzy models, lasso, ridge, regression trees, different variants of neural networks.

3.3.3. Regression with Neural network

The solution for the regression problem we found in neural networks [27] with their ability to solve the regression problems. However, we had to propose some additional modifications besides classical feed forward NN [28] tuning performed during NN engineering, such as number of layers and neurons, and selection of training algorithm.

We based on the classical feed-forward NN, which performs well for the regression of the position of a marker. The residuals should reveal distortions if the NN is not over-trained. However, fluctuations of the residual which resemble pink noise, can cause false detections when thresholded. Therefore

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

we decided to mimic multistart NN training, with P-fold replication of the target output values $Y_p = [x, y, z]$ values. So our prediction has a final step:

$$\hat{Y} = \frac{1}{M} \sum_{m=1}^{M} \hat{Y}_m. \tag{10}$$

Random valued initialization with scaled conjugate gradient training method make each of the output replicas follow the true values, and the errors (residuals) are not correlated, unless they represent actual distortion. Hence, their averaged residual values exhibit much lower noise level so it allowed us to reveal the slow artifacts with thresholding.

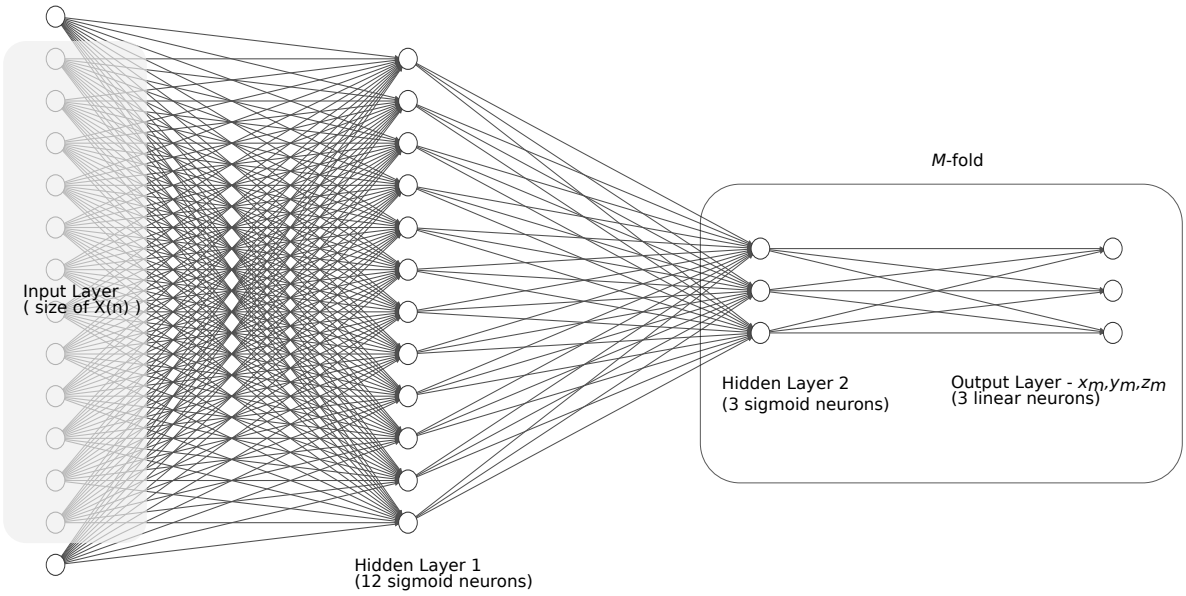

Figure 8. Architecture of neural network used for regression

The design of NN structure is a kind of art, as there are no unambiguous rules or guidelines. Usually, it requires to simulate with parameters sweeping for a domain of possible (feasible) numbers of layers and neurons, with critical review of obtained performance (MSE or classification ratio) [29]. We shared that approach and reviewed the performance of NN using the test data. Finally, the architecture NN we employed is presented in Fig. 8. It is ordinary fully connected FFNN, with 2 hidden layers – first containing 12 sigmoidal neurons, second containing $4 \cdot M$ sigmoidal neurons. The output is 3 valued x, y, z vector replicated P times – we decided to use 5 fold replication. As an input we used similar set of neighbor and parent coordinates to Eq. 9, additionally we enhanced it with moving average of own value of a marker. The latter could make the NN to follow momentary slow changes, therefore the window of a moving average (MA) should be notably larger then length of detected distortions (we assumed it to be 200 samples). By extensive testing we identified also number

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

previous values (= 1) and order of power used to raise the input data ($L = 2$). Finally each input vector X is long and is assembled of certain parts as given below:

$$X(n) = \left[\begin{array}{c} \overbrace{x_p(n), y_p(n), z_p(n), x_p(n-1), \cdots, z_p(n-k)}^{\text{current value and } k \text{ former of parent marker } (p)}, \\ \overbrace{x_{s1}(n), y_{s1}(n), z_{s1}(n), x_{s1}(n-1), \cdots, z_{slast}(n-k)}^{\text{current value and } k \text{ former of first..last siblings}}, \\ \vdots \\ \underbrace{x^L_{s1}(n), y^L_{s1}(n), z^L_{s1}(n), x^L_{s1}(n-1), \cdots, z^L_{slast}(n-k)}_{\text{current value and } k \text{ former of first..last siblings raised to } L\text{th power}}, \\ MA_x(n), MA_y(n), MA_z(n) \end{array} \right]. \tag{11}$$

In Fig. 9 we demonstrate the prediction result for real sequence contaminated with artificially introduced distortion. We can clearly observe, that NN residuals contain expected changes in signal, whereas residuals from polynomial model are inconclusive.

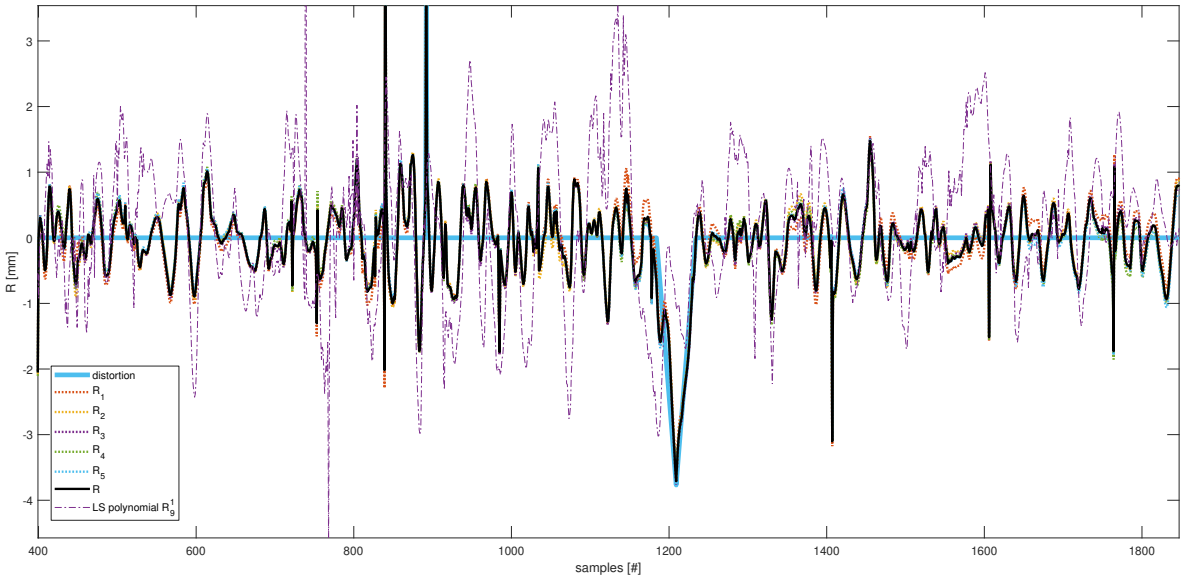

Figure 9. Actual distortion prediction and residuals – component (R_m) and final (R), compared with polynomial residual R^1_9

3.4. Recognition and classification of distortions

The detection works using the prediction residuals – therefore we can observe the deviations which cannot be explained by expected movement of markers described with model. To some extent, these deviations can be considered as an innovation in the marker position, which results in position change beyond prediction. Therefore minor residual values can be interpreted as normal motion, whereas large or sudden changes imply artifacts. Knowledge of statistical properties of residuals allow us to evaluate thresholds for detection outlying of the normal variability of residuals.

Detection process (see Fig. 4) is organized into strict pipeline consisting of four stages, where we detect distortions from the simplest to the hardest to detect with removing the detected distortion through interpolation after each stage. Such an approach ensures proper classification, otherwise we would get false positive classification due to fact that subsequent methods can be also sensitive to simpler distortions f.e. slow change would be also sensitive to rectangular distortion if the amplitude of distortion would be sufficient.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

3.4.1. Locating sudden changes

Sudden changes are well detectable in the derivative of basic signal meanwhile slow changes require use of a base representation of residuals to measure the deviation. For the approximation of derivatives we use differentials:

$$\Delta X(n) = X(n) - X(n-1). \tag{12}$$

Discrimination of different types of sudden changes cannot rely on differentials only. It is so because of fact, that a strong peak in ΔX notifies about the existence of a sudden change, but it does not bring information about the structure - the duration of the change or its neighborhood. Therefore we employed mathematical morphology methods (MM) for analysis of shape of those sudden changes. We used the following MM operations:

$$\text{Erosion: } E(n) = x(n) \ominus S = \max(\forall_{j \in S_n} x(n-j)), \tag{13}$$

$$\text{Dilation: } D(n) = x(n) \oplus S = \min(\forall_{j \in S_n} x(n-j)), \tag{14}$$

$$\text{Opening: } O(n) = x(n) \circ S = (x(n) \ominus S) \oplus S, \tag{15}$$

$$\text{Closing: } C(n) = x(n) \bullet S = (x(n) \oplus S) \ominus S, \tag{16}$$

$$\text{Top-hat: } T_w(x(n)) = x(n) - (x(n) \circ S), \tag{17}$$

where: $x()$ is a 1D signal, S is structuring element defining points to be taken into consideration (j), S_n is structuring element centered (translated) at n

An additional morphological method is seeking sudden changes, we implemented a scanning function (`find_derivate_pairs(dX,T,maxlen)`), which is looking for opponent differential pairs `dX` exceeding threshold level `T` and being no more distant than some presumed maximal length `maxlen`. It results in binary decision variable marking located ranges. The function parameters – threshold and distance would depend on the data characteristics and sampling frequency.

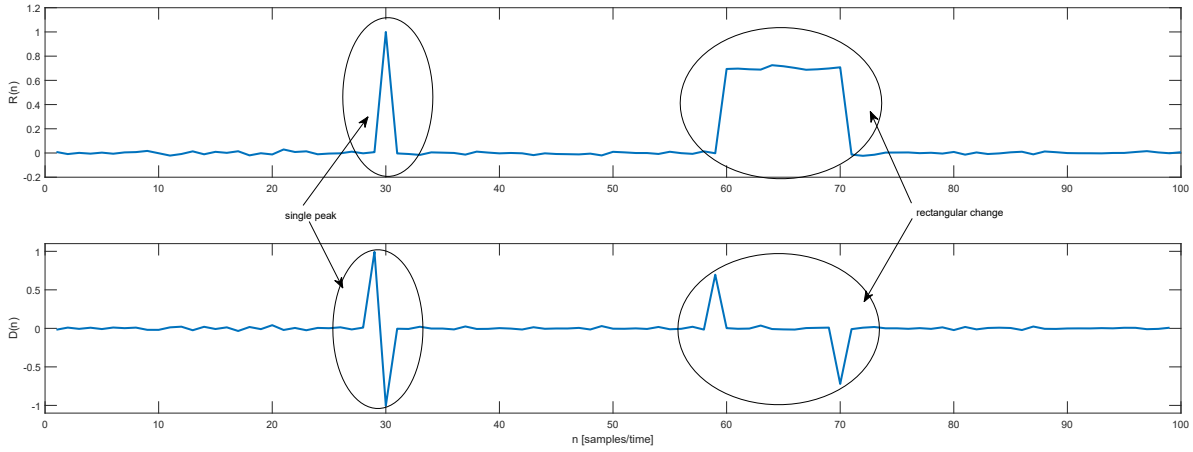

Figure 10. Appearance of sudden distortions in residual/high-pass ($R(n)$) – single peak and step change, and their corresponding peak pairs in differential $D(n)$

3.4.2. Identifying single peaks

It is the first stage of processing. Single peak and heavy noise are two appearances of the same short-term distortion – the key difference is that single peaks are isolated within some neighborhoods, whereas, in heavy noise segments, numerous peaks occur next to each other. To identify the isolated peaks, we employed the following sequence of operations.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

First, we remove low frequencies (presumed to be legitimate) using median filter::

$$X_{HP} = X - \text{median}(X, window), \tag{18}$$

where *window* size should be larger several times than maximal peak length.

Then, we calculate differential:

$$D_{HP}(n) = X_{HP}(n) - X_{HP}(n-1), \tag{19}$$

which is cleaned of non-interesting low values using thresholding :

$$\tilde{D}_{HP}(n) = \begin{Bmatrix} D_{HP}(n), & \text{if } D_{HP}(n) > threshold_1 \\ 0, & \text{otherwise} \end{Bmatrix} \tag{20}$$

Next, the identification of probable peaks is based on the assumption that their differential sums locally to a small value (in theory ≈ 0), whereas the local sum of absolute values of the differential is high. So we calculated these sums within window W_n:

$$movSum(n) = \sum_{j \in W_n} \tilde{D}_{HP}(n-j) \tag{21}$$

and

$$movSumAbs(n) = \sum_{j \in W_n} |\tilde{D}_{HP}(n-j)| \tag{22}$$

which are then tested simply as:

$$ampRatio(n) = \frac{movSum(n)}{movSumAbs(n)} \tag{23}$$

when the *ampRatio* is larger than the assumed threshold (we assumed 5) it implies there is a peak candidate:

$$peakCandidates(n) = \begin{Bmatrix} 1, & \text{if } ampRatio(n) > threshold_2 \\ 0, & \text{otherwise} \end{Bmatrix} \tag{24}$$

Finally, to identify single peaks only, we employed binary top-hat, which rejects peaks within the neighborhood defined by the structuring element. It allowed us to keep isolated peaks only:

$$peaks = T_w(peakCandidates, S) \tag{25}$$

Tunable parameters of the stage are:

- *window* for moving average, we assumed it to be 19 samples long,
- $threshold_1$ which is calculated statistically from the data using $k \cdot \sigma$ of D_{HP} – w employed $3 \cdot \sigma$ as a default value, however any k can be provided as parameter,
- $threshold_2$ (anti-sensitivity) for $ampRatio$
- *maxSize* which declares maximal size of expected peaks, it affects size of moving sums windows W_n, which is $2 \cdot maxSize + 3$ samples long, it also defines size of linear structuring element for morphological operations S.

3.4.3. Heavy noise

The heavy noise detection is somewhat similar in general design to isolated peaks detection, but there are also differences in the details. Foremost, we assume that input data are already clear of isolated peaks. First we calculate the differential:

$$D_1 = X(n) - X(n-1), \tag{26}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

from which we remove low frequencies using high pass variant of Savitzky-Golay filter (Eq 7):

$$D_{1\_HP} = SavitzkyGolayHiPass(D_1, L, M). \tag{27}$$

Next, we remove small fluctuations within prospective areas of high value in D_hp with morphological closing (float):

$$D_{1\_HP\_cleaned} = |D_{HP}| \bullet S. \tag{28}$$

These values are now thresholded:

$$rawNoiseAreas(n) = \begin{Bmatrix} 1, & \text{if } D_{1\_HP\_cleaned} > threshold \\ 0, & \text{otherwise} \end{Bmatrix}. \tag{29}$$

Raw noise areas are finally cleaned by removing holes with use of morphological closing, but binary variant this time:

$$heavyNoise = rawNoiseAreas \bullet S. \tag{30}$$

Finally, heavy noise segments shorter than *minLen* attribute are rejected.
Tunable parameters of the heavy noise detection stage are:

- *threshold* which is calculated statistically from the data using $k \cdot \sigma$ of D_{HP} – w employed $2 \cdot \sigma$ as a default value, however any k can be provided as parameter,
- minimal length of segment (*minLen*), which is used to define linear structuring element S as $2 \cdot minLen - 1$, we assumed $minLen = 20$ samples ,
- default parameters of Savitzky-Golay are L=5, M=13.

3.4.4. Step changes

Step changes is another differential-based detection, which resembles the two former detections. It requires removing isolated peaks and heavy noisy areas. After that, identification of rectangular alike changes becomes a simple problem.

The first two steps are shared with heavy noise detection. We perform differential computation(D_1), which is then high pass filtered with Savitzky-Golay filter, so we get D_{HP}. Next, we employ simple scanning with `find_derivate_pairs`, which seeks for the areas between the complementary pairs of differential peaks (above *threshold*), which we interpret as a rectangular distortion, it is visible in Fig. 10. This scanning requires setting up two parameters – *minLen* and *maxDist*, identifying the minimal length of step change and maximal distance of searching.

Tunable parameters for the step change detection stage are:

- *threshold* which is calculated statistically from D_{HP} using $k \cdot \sigma$ of – we employed $3 \cdot \sigma$ as a default value, however any k can be provided as parameter,
- minimal length of segment (*minLen*), we assumed $minLen = 20$ samples,
- maximal searching distance *maxDist*, we assumed 200 samples as default value,
- default parameters of Savitzky-Golay are the same as for heavy noise L=5, M=13.

3.5. Identifying slow changes

The slow change of position is a class of distortion notably different than the other ones. It requires a separate approach for detection, due to the fact, that its nature makes it hard to detect with differential analysis. We decided to employ a model-predictor, which predicts proper marker position on the basis of its neighbor markers (parent and sibling), the deviations from such a model (R – residuals) are analyzed looking for notably large and long deviations which are identified as artifact hills or valleys. Alas, the hills and valleys can be of a different scale, so both their length and differential can vary significantly. Moreover, inaccuracies of a predictor at the turning points can also seem similar to short-term slow change (of small amplitude) appearing when the predictor cannot

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

follow the change of value. Though, based on the statistical properties of the residual of prediction and on the fact that we know that the distortion should be rather long (as it is accumulated reconstruction error), one can make certain assumptions allowing detection of the distortions.

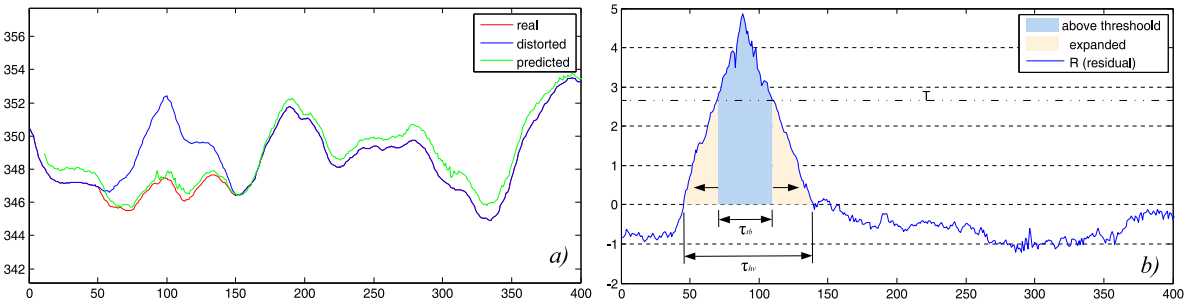

Figure 11. Slow detection: a) original, predicted and distorted signal, b) residual with hysteresis thresholding

We decided to use scanning of values of the regression residua with hysteresis thresholding. It can be described with the following, heuristics based steps:

1. The upper threshold T_u was set up with a $K \cdot \sigma$ rule of a thumb – in our case, three times σ_R was selected as default would identify the significant tops and bottoms of hills and valleys.
2. If the top or bottom length is shorter than some minimal τ_U we skip it (0.2 s - 20 frames in our case) assuming to be short term fluctuation
3. After the identification of a top/bottom value we are looking to find the rest of a distortion (below threshold) - the marked range expands both sides iteratively (in the past and future) until the residual value goes below/above lower threshold T_L (0.5 in our case).
4. If the overall located distortion (hill/valley) is shorter than some value τ_{hv} (0.5s) it is omitted as one can consider it as a short-term fluctuation of the predictor.

4. Verification of the method

The verification of the efficiency of the proposed approach is three-fold. In the first experiment, we analyzed the efficiency of distortion classification using synthetically generated distortions in artifact free sequences, which allowed us to provide some statistics on the classification efficiency. In the second test, we compared the performance of the proposed approach to the human operators of various experiences – from novice to expert one. The last test is connected with the applicability of the artifact classification for the data cleaning with a pool of generic reconstruction algorithms.

4.1. Materials and methods

4.1.1. The data

For testing purposes, we used data set acquired for professional applications in the motion capture laboratory. The sequences were obtained at PJAIT human motion laboratory using the industrial-grade Vicon MX system. The system capture volume is 9 m x 5 m x 3 m. To minimize the impact of external interference like infrared interference from sunlight or vibrations, all windows are permanently darkened and cameras are mounted on scaffolding instead of tripods. The system is equipped with 30 NIR cameras manufactured by Vicon – 10 pieces of each kind: MX-T40, Bonita10, Vantage V5.

During the recording, we employed two system configurations – a standard animation pipeline, where data were obtained with Vicon Blade software using 53 marker setup and a typical biomechanic setup using Vicon Nexus software using 39 markers setup. The trajectories were acquired at 100Hz and by default they were processed in a standard, industrial quality way, which includes manual data reviewing, cleaning and denoising, so they can be considered to be distortion free. However, depending on the experiment, different variants of the recordings were used in experiments; these

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

are raw unprocessed data, processed (cleaned), and artificially modified variant with controlled amount and locations of distortions. Information, on which variant is used, is provided in the detailed description of experimental protocols.

Table 1. List of mocap sequence scenarios used for the testing

No.	Name	Scenario	Duration	Difficulty
1	Static	Actor stands in the middle of scene, looking around and shifting from one foot to another	22 s	easy, static
2	Sitting	Actor stands in the middle of scene, then sits on a chair, after few seconds stands again, repeats this three times	29 s	occlusions

4.1.2. Experimental protocols

We planned the first experiment (E1) to test the performance of the proposed method for a controlled dataset, with a perfectly clean sequence and controlled artificial distortions. It involved the first recording from the Tab. 1 – *static*, which was manually cleaned by an expert, so it is artifact-free ground truth. Next, we introduced distortions into them at random locations (randomly drawn markers) and random amplitudes – see noise contamination procedure given further in sec. 4.1.3. We kept track of the distortions and their types, therefore, we were able to verify whether the error classification is correct. The criteria for an evaluation in the artifact recognition task are pretty straightforward, they are classification rates (true and false recognition), presented as a confusion matrix. The simulation was performed for three distortion shares 5%, 10% and 20%; each was executed 1000 times and the results are aggregated as averaged confusion matrices (rounded). For each class, we calculated the following measures:

$$\text{sensitivity (true positive rate)} : TPR = \frac{\text{tp}}{\text{tp}+\text{fn}} \cdot 100\%, \tag{31}$$

$$\text{miss} - \text{rate (false negative rate)} : FNR = \frac{\text{fn}}{\text{tp}+\text{fn}} \cdot 100\%, \tag{32}$$

$$\text{fall} - \text{out (false positive rate)} : FPR = \frac{\text{fp}}{\text{tp}+\text{fp}} \cdot 100\%, \tag{33}$$

$$\text{precision (positive predictive value)} : PPV = \frac{\text{tp}}{\text{tp}+\text{fp}} \cdot 100\%, \tag{34}$$

$$\tag{35}$$

Additionally, two more measures were employed to quantify performance. F-score is a scalar describing efficiency of overall classification for all classes [30] – its values are between 0 (no proper classification) and 1 (perfect classification). From various equivalent formulas, we chose the following one, because it is simple to adapt to the multiclass problem:

$$F = \frac{\text{tp}}{\text{tp} + \frac{1}{2}(\text{fp}+\text{fn})} \quad . \tag{36}$$

The other measure was Matthews Correlation Coefficient (MCC) [31], which is a quality measure intended for characterizing the classification efficiency for imbalanced populations of classes (as in our cases). Its values scale between -1 for no classification and 1 for perfect classification. The formula is given as:

$$MCC = \frac{\text{tp}\cdot\text{tn} - \text{fp}\cdot\text{fn}}{\sqrt{(\text{tp}+\text{fp})(\text{tp}+\text{fn})(\text{tn}+\text{fp})(\text{tn}+\text{fn})}} \quad , \tag{37}$$

where cardinality of classifications are denoted as: tp – true positive classifications, fp – false positive, tn – true negative, fn – false negative.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

The second experiment (E2) involved a comparison of performance to the four operators of the mocap facility with different levels of experience - two beginners (2 and 3 months experience), one intermediate (1,5 years experience), and one expert (10 years experience). The test is intended to be qualitative verification of the proposed approach and to verify the proposal using real life data. We used the raw forms (not cleaned) recording that contains all the kinds of distortions. Such data were used against the proposed detection algorithm. Apart from automatic processing, all four operators did normal data screening and cleaning as well. These manual processing steps, using Vicon Nexus software, were a standard approach in the lab, that is in everyday usage in the facility. In the final step, the results obtained by the algorithm and four operators were reviewed by an expert – a human mocap system operator with long experience in data editing and cleaning. The results are reported as raw numbers of distortions located, compared, and verified against human judgment.

The last experiment (E3) is the verification of the applicability of the proposed approach. It is intended to be rather a proof-of-concept of targeted distortion cleaning. Therefore, it uses different variants of static person sequence with the distortions of variable intensity and duration introduced into the recording – taking 5%, 10%, and 20% of overall length – similarly to E1. In the tests, we employed the following reconstruction methods: Savitzky-Golay (13th order polynomial over 101 samples window), linear interpolation, spline interpolation, and FFNN prediction (as given in p. 3.3). Each of the methods is applied in the locations of detected artifacts only, the rest of the signal remains intact. All distortions were simulated separately in this case, with a randomized: location (marker), time, duration, and amplitude with 10 mm average value and 4 mm std deviation. The simulations of contamination-detection-reconstruction were performed 200 times, for each fold we obtained a quality measure, which finally was averaged. We assumed root-mean-square error (RMSE) as the measure of quality, it was computed over all the coordinates and samples in the considered sequence:

$$RMSE = \sqrt{\frac{1}{K \cdot N} \sum_{k=1}^{K} \sum_{n=1}^{N} (\hat{x}_k(n) - x_k(n))^2} \quad , \tag{38}$$

where: K is a number of variables in timeseries, N is a number of samples, $x()$ is original value, $\hat{x}()$ is reconstructed value.

4.1.3. Artifact contamination procedure

The procedure of distortion contamination, which was employed in E1 and E3, introduces artifact distortions into the sequences in a controlled way – we log types, duration, locations, and amplitudes of distortions. The contamination can include or each distortion separately, either mixture of all kinds in equal proportions. The key parameter characterizing the experiment is a time share (distorted time fraction) for which distortions are generated. For the interpretation clarity, we ensure that there might occur only one distortion at a time, therefore time share denotes that at a given fraction of time there occurs a distortion. The sequence of distorting the signal is as follows: at first, we draw locations to contaminate with 'bulk' distortions – slow, step changes, and heavy noise, next we seed randomly isolated peaks. Distortion parameters are set up randomly, for each instance of distortions:

- sign is +1/-1 value drawn with equal probabilities,
- amplitude is a Gaussian random variable with assumed amplitude and standard deviation (in the tested cases, hey are $\mu = 5$ or 10 mm and $\sigma = 0.4 \cdot \mu$), these values are used to scale the peak of rectangle or triangle distortion and as a standard deviation in heavy noise area,
- distortions duration and intervals is a Poisson process, an average length of distortion set up to 50 samples, and the interval length is adjusted according to the duration of the sequence and the target amount of the given distortion.

An important remark here is that the distortions introduced are quite demanding for the detection procedure. The amplitudes are on average small (5 and 10 mm) and of short (0.5 sec) duration, therefore

 doi:10.20944/preprints202203.0249.v1

we can assume that the synthetic tests are rather rigorous and more difficult to detect than in real life scenarios

4.2. Results and discussion

The outcomes of all three experiments are provided in the successive sub paragraphs. They are accompanied by interpretation and discussion of results.

4.2.1. Synthetic distortion classification

The outcomes of classification for synthetic noise are demonstrated in Figures 12 and 13 for average amplitudes 5mm and 10mm respectively. The raw results in confusion matrices demonstrate average number of samples (rounded towards whole sample) assigned to specific classes (correct or not) for 1000 simulations. They present averaged confusion matrices for three distortion shares 5%, 10% and 20% of overall time of sequence with two average amplitudes - 5mm and 10mm. According to the length of the recording, in simulation the contamination procedure should produce approximately 160, 320, 640 distorted samples respectively of each distortion type, they should be in equal proportions.

Regardless of the number of distortions, the results are pretty consistent, they were also very similar for numerous additional simulation runs which are not included here. The fractions of true and false classifications hold across the runs, the same holds for F-scores, that are above 0.999 for the amplitudes and distortion shares (and also holds for MCC but to a lesser extent) – all these large values are thanks to the proper classification of the most numerous clear signal samples. Each specific class requires separate insight into the results. These are as follows:

- clear signal is identified properly for more than 99% of samples; a negligibly small amount of distorted samples are erroneously identified as clean signal (comparing to overall cardinality of the class).
- for peak changes sensitivity is approximately 66% and 90%, and the main misclassification is into a clear signal; this class is not a cause of confusion for other classes than clean signal with usually below $FPR = 50\%$
- for heavy noise sensitivity is above 88%, and main confusions are step change and clear signal; this class is rarely erroneously recognized in place of others ($FPR = 4 - 8\%$), the main confused class is a clear signal
- for step changes sensitivity is approximately 70% and main confusion is slow change; this class is erroneously recognized in place of others at a moderate rate ($FPR = 12 - 27\%$) – here clean signal and heavy noise are wrongly identified,
- for slow change, TPR is a bit more than 50% of the and main confusion is clear signal; this class is difficult and erroneously recognized in place of others very often ($FPR = 80 - 90\%$) – usually, it is clear signal, but also step change and heavy noise are wrongly identified.

Considering the above results, the proposed method has sensitivity from moderate to high (depending on the class), and also quite high precision for all classes but one (slow change). What is notable from the practical point of view, false negative detection (having relatively small values) is way more undesirable than others. False positive, or detecting a wrong class of the distortion, still would result in pointing out the operator to the potential error location or in case of automatic error filtering would cause using of a repairing procedure (see 4.2.3).

The difficulty in identification of slow change was expected, it comes from the fact that this change can be subtle, and poorly distinguishable in predictor residuals, which resemble pink noise in our case. The latter is also a cause of high FPR. We analyzed alternative regression methods as a model – neural networks (simple FFNN and NARX-NN models), ridge, lasso, and SVR. However, the results were either poor or impractical (due to long training time), or both. On the other hand, false positive error (quite frequent) is of much lesser importance than a false negative, the former might result in suggesting additional locations to the operator for reviewing, or using the interpolating method which

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

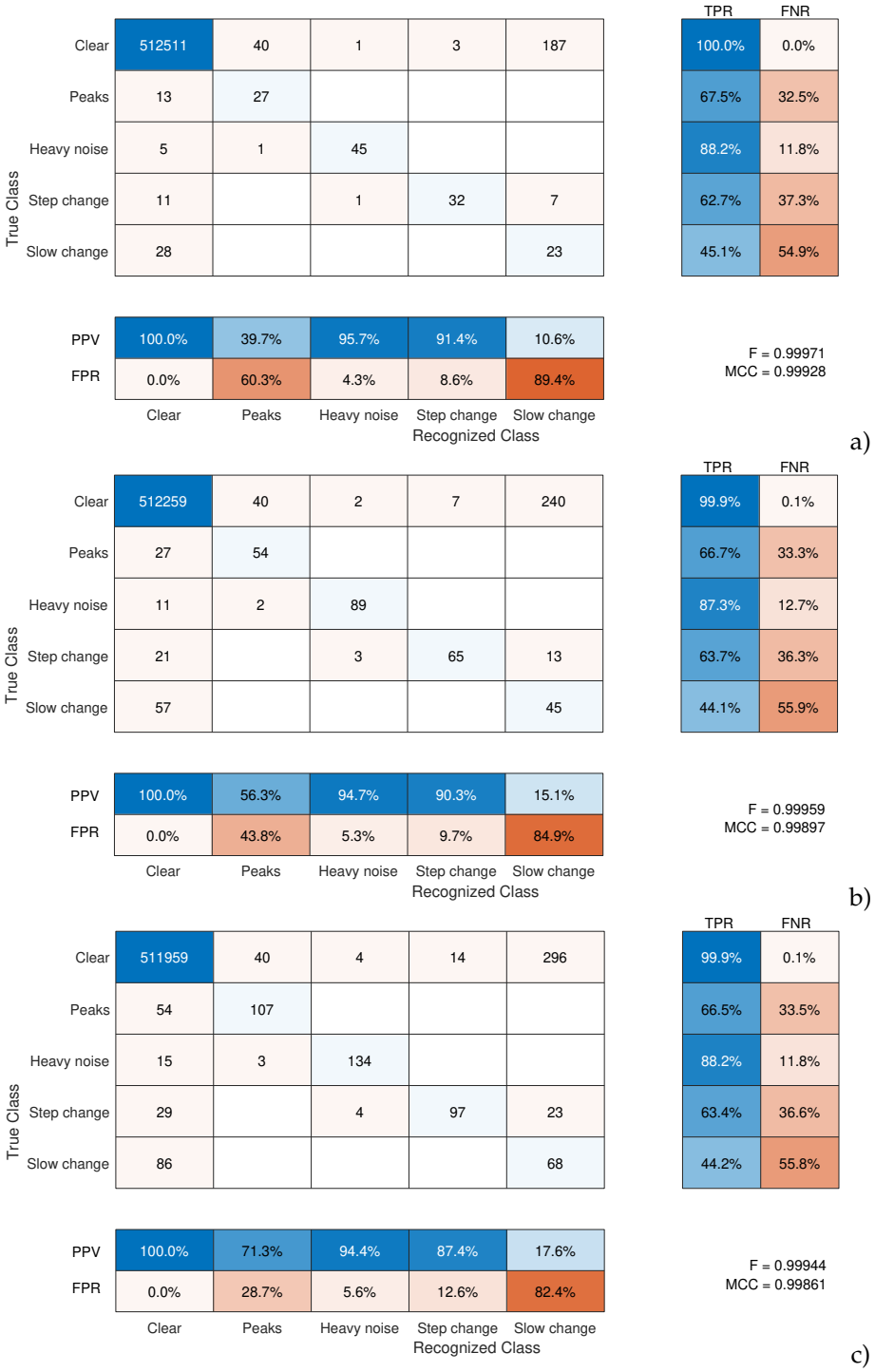

Figure 12. Averaged confusion matrix for detection of synthetic noises for 1000-fold simulation with 5 mm average amplitude of distortions and share: a) 5%, b) 10%, c) 20% of time

should not degrade the signal significantly; whereas the latter might result in preserving the distortion in the signal.

4.2.2. Comparing to the human operators

Table 2 comprises the numbers of detected distortions in the recording processed in a standard pipeline, and the recording processed by each of four operators. The detected distortions are compared and verified by a human operator, who either approved the classification or rejected them.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

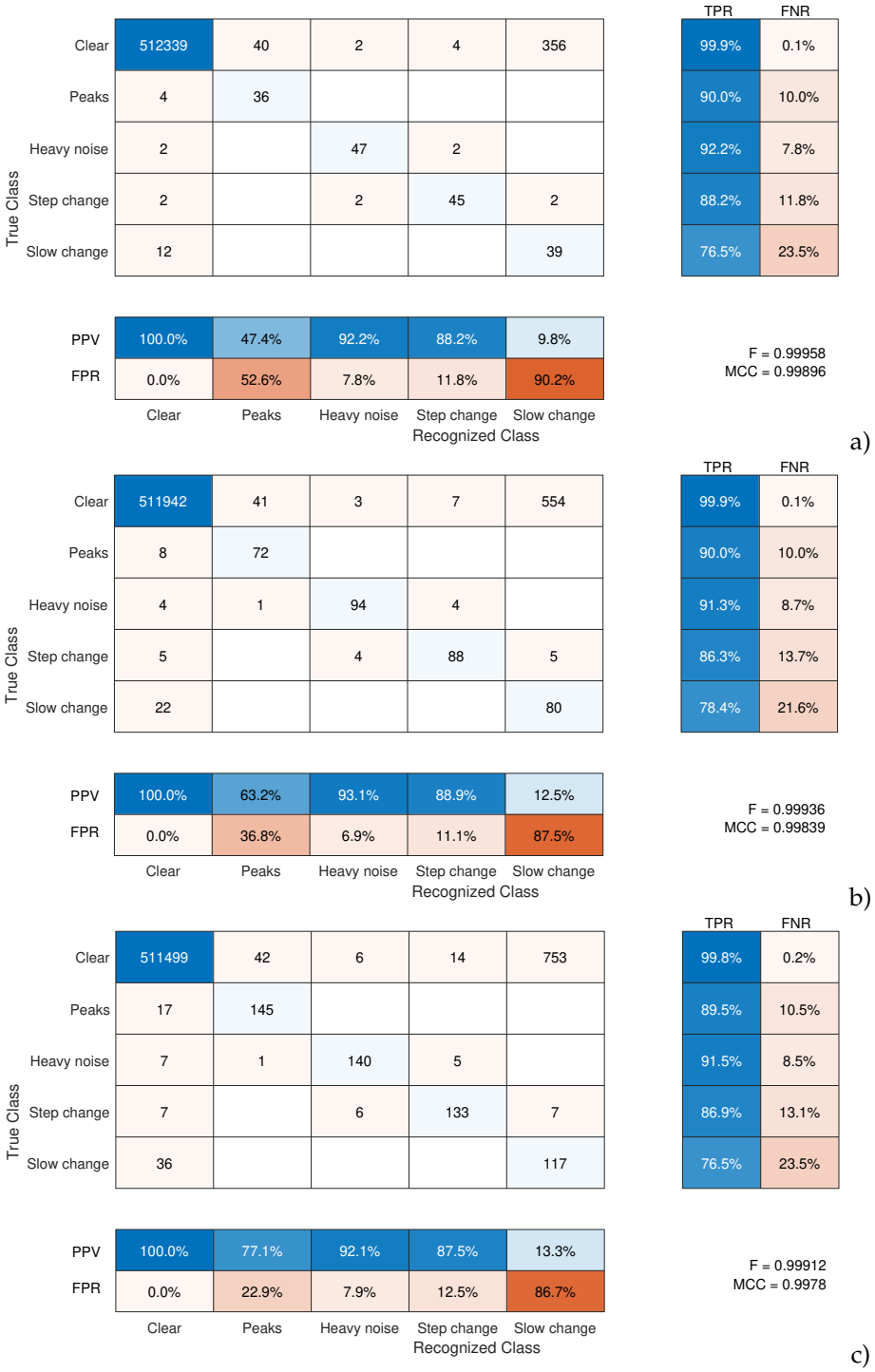

Figure 13. Averaged confusion matrix for detection of synthetic noises for 1000-fold simulation with 10 mm average amplitude of distortions and share: a) 5%, b) 10%, c) 20% of time

In the recording processed by the machine, the proposed algorithm found 29 errors - peaks, step, and slow changes and 16 of these errors were correctly classified. The algorithm classified a sudden, very dynamic hand movement in a relatively static recording as slow changes (13 incorrectly classified errors). In addition, the algorithm did not detect 4 errors, after careful analysis, it turned out that these errors do not belong to any of the previously defined classes - they are a combination of a single peak and a slow change. An example of such an error is shown in the figure 14.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

Table 2. Comparing amount of distortions located by proposed method to the human operator (E2)

Operator	Seq. No	Recording	Errors Identified by		Errors Verification		
			Human	Algorithm	Approved	Rejected	Missed
None	2	Sitting	20	29	16	13	4
Expert	2	Sitting	20	9	0	0	0
Intermediate	2	Sitting	18	11	2	9	0
Beginner 1	2	Sitting	10	37	20	17	2
Beginner 2	2	Sitting	11	46	26	20	1

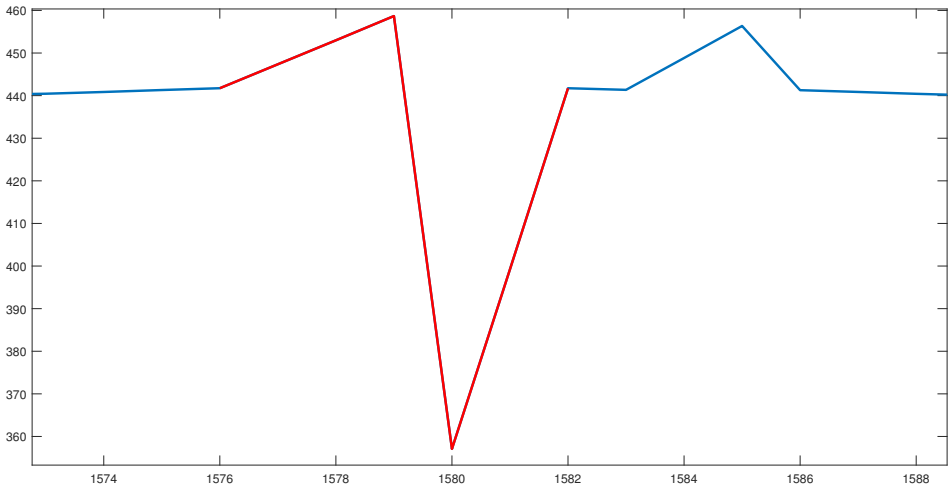

Figure 14. Additional combined distortion type

In the recording processed by the expert, the algorithm again incorrectly classified the hand movement as slow changes. The result was similar for the intermediate operator, with the difference that the algorithm found two errors omitted by the human (two small peaks).

In the case of a recording repaired by beginners, both the algorithm and the expert found more errors after the repair than before. This is due to the selection of an inappropriate method of repairing a given artifact. For example, when the distortion occurred only on one axis, the person, in order to correct the error, removed the marker trajectory for those few frames when the error occurs. This resulted in the creation of an additional gap, which the beginner operator fills using simple interpolation. In the case of longer artificial gaps, interpolation caused the data to be incorrectly reconstructed, and the error no longer appears on one axis, but on all three.

4.2.3. Applicability testing

The results are gathered in Tables 3-5. Each field presents the averaged RMSE of 200 fold repeated simulation process of distorting the test sequence and its reconstruction using various procedures. Each distortion type was considered separately. It allowed us to quantify how each reconstruction method reduces the distortions. Fig. 15 illustrates the reconstruction results – demonstrating the ground truth, distorted signal value, and outcomes of four variants of reconstruction. In the tables for each distortion type and reconstruction method, we have two RMSE values to compare – for hypothetical perfect classification, and for actual classification with the proposed algorithm.

In the results, we recognize the different efficiency of the tested reconstruction methods for different distortions. We could also clearly observe that the efficiency of detection of artifacts directly affects the ability to restore the signal. The key observations are summarized in a few points:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

Table 3. RMSE after reconstruction with different methods (with perfect and algorithmic artifact classifications) for mocap sequence with 5% distorted time in sequence

	Peaks	Heavy Noise	Step change	Slow change
Distorted	0.19065	0.17717	0.18069	0.10129
Linear interpolation (perfect)	0.00136	0.18322	0.15939	0.18795
Linear interpolation (classified)	0.03947	0.18514	0.15623	0.29339
Savitzky-Golay filter (perfect)	0.01900	0.04963	0.17440	0.10130
Savitzky-Golay filter (classified)	0.08159	0.06855	0.17553	0.10173
Spline interpolation (perfect)	0.00025	0.11041	0.09780	0.10972
Spline interpolation (classified)	0.03429	0.68692	0.10895	0.19935
FFNN predictor (perfect)	0.01841	0.01953	0.01933	0.01875
FFNN predictor (classified)	0.03944	0.04547	0.04409	0.11000

Table 4. RMSE after reconstruction with different methods (with perfect and algorithmic artifact classifications) for mocap sequence with 10% distorted time in sequence

	Peaks	Heavy Noise	Step change	Slow change
Distorted	0.26906	0.26340	0.26282	0.15037
Linear interpolation (perfect)	0.00186	0.25487	0.28409	0.27662
Linear interpolation (classified)	0.05144	0.26360	0.27618	0.38908
Savitzky-Golay filter (perfect)	0.02678	0.07211	0.25361	0.15046
Savitzky-Golay filter (classified)	0.08959	0.08895	0.25510	0.15083
Spline interpolation (perfect)	0.00039	0.14679	0.15207	0.15955
Spline interpolation (classified)	0.04754	0.95512	0.16358	0.23918
FFNN predictor (perfect)	0.02785	0.02468	0.02581	0.02612
FFNN predictor (classified)	0.05537	0.05201	0.06349	0.14717

Table 5. RMSE after reconstruction with different methods (with perfect and algorithmic artifact classifications) for mocap sequence with 20% distorted time in sequence

	Peaks	Heavy Noise	Step change	Slow change
Distorted	0.38228	0.39623	0.39247	0.21676
Linear interpolation (perfect)	0.00273	0.44107	0.44015	0.39172
Linear interpolation (classified)	0.07007	0.45679	0.45630	0.55692
Savitzky-Golay filter (perfect)	0.03880	0.11228	0.37892	0.21704
Savitzky-Golay filter (classified)	0.10383	0.16007	0.38120	0.21748
Spline interpolation (perfect)	0.00058	0.24337	0.28188	0.25274
Spline interpolation (classified)	0.06753	1.58979	0.37534	0.35679
FFNN predictor (perfect)	0.04169	0.03711	0.03972	0.03735
FFNN predictor (classified)	0.07903	0.11433	0.10284	0.20549

- peak changes are effectively removed with interpolation methods – simple linear or spline (piecewise cubic polynomial); the other two methods in perfect detection wouldn't offer even comparable efficiency, yet in actual classification, they offer just slightly worse performance,
- FFNN offers the best performance for all the 'bulky' distortions (of longer duration), both hypothetical and classified cases
- heavy noise, aside FFNN, is well cleaned also with Savitzky-Golay filter (see Fig. 15b)
- step changes can be effectively removed with FFNN only,
- slow changes are the most contradictory – the only appropriate reconstruction method is FFNN, in the case of perfect detection the efficiency is high, but due to limited actual detection, the results are quite poor. These results correspond well to the detection of slow changes in E1 – low sensitivity and high fall out.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 17 March 2022 doi:10.20944/preprints202203.0249.v1

The above outcomes of reconstruction are just preliminary results. It should be further analyzed in a separate study for the possible reconstruction methods involving other predictors, interpolations, rigid body model, and projections on geometric constraints.

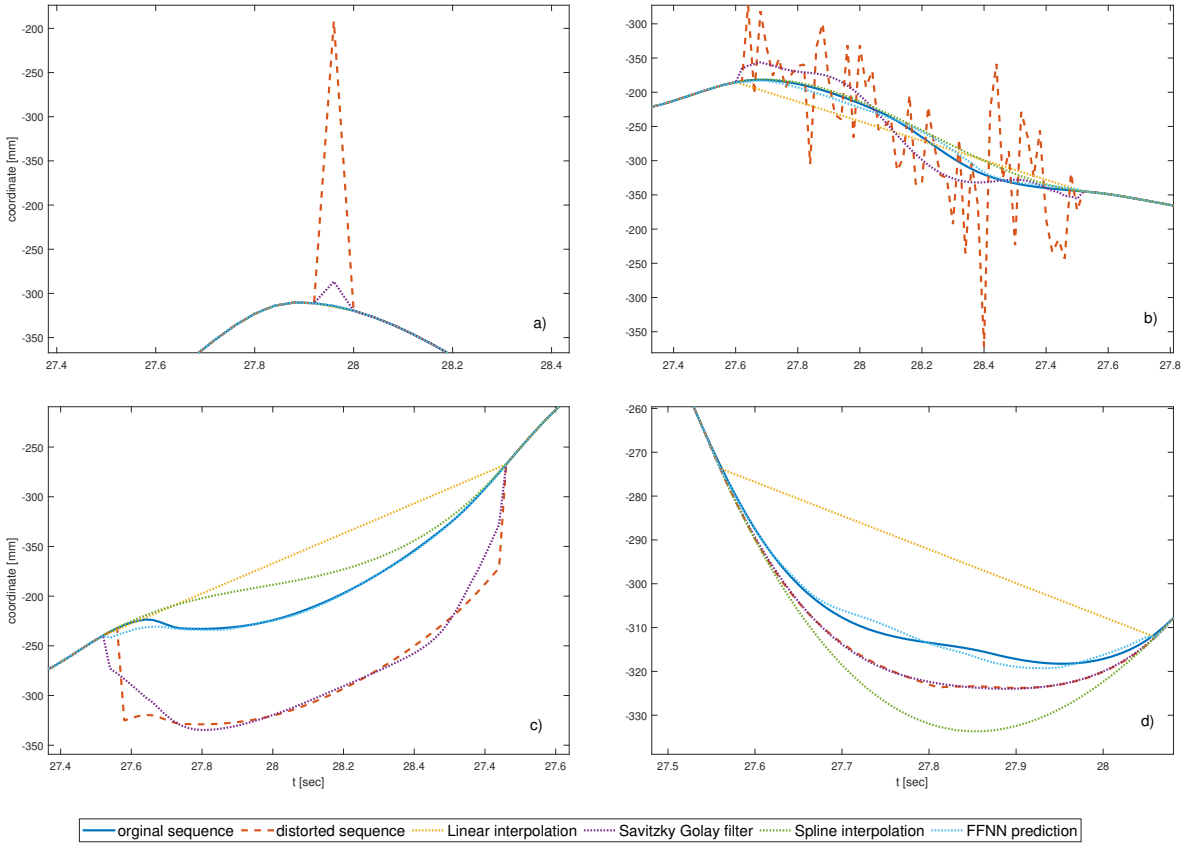

Figure 15. Artifacts and their removing with methods tested in E3: a) single peaks, b) heavy noise, c) step change, d) slow change. Please mind various scales in axes.

5. Summary

In this article, we addressed the issue of artifacts occurring in the mocap signal. We proposed the method for their detection and demonstrated how to employ the detection method for improvement of the signal fidelity. The method proposed in this article seems to be quite effective for sudden changes, and it can detect distortions of relatively small amplitudes. As for the slow changes, its outcomes are moderate, since we observe a relatively large number of false positive detections. However, we expected that this class of distortions might be difficult to detect. This topic is worthy of further studying.

Comparing to the human operators, the proposed solution cannot outperform experienced professionals, however, it offers notably better performance than novice ones. On the other hand, even for the expert, it can save time by suggesting locations for reviewing.

The proposal is possible to be adopted in currently existing software as an optional step of signal refinement and/or for automatic support for the mocap sequence editors. Further improvements are still possible, but require additional research like employing better predictive models. Also, the engineering approach could be beneficial for detection efficiency. One of such possible improvements could be to detect distortions for all the three coordinates of a marker jointly, since distortions usually occur in more than a single coordinate. Also, studying the reconstruction methods remains the topic that we plan to investigate in the future.

Author Contributions: conceptualization, P.S.; methodology, P.S., M.P.; software, P.S., M.P.; investigation, P.S., M.P.; resources, M.P.; data curation, M.P.; writing–original draft preparation, P.S., M.P.; writing–review and editing, P.S., M.P..; visualization, P.S., M.P.

Funding: The research described in the paper was performed within the statutory project of the Department of Graphics, Computer Vision and Digital Systems at the Silesian University of Technology, Gliwice (RAU-6, 2022). APC were covered from statutory research funds.
M.P. was supported by grant no ***grant number comes here***

Acknowledgments: The research was supported with motion data by Human Motion Laboratory of Polish-Japanese Academy of Information Technology.

Conflicts of Interest: The authors declare no conflict of interest. The funders had no role in the design of the study; in the collection, analyses, or interpretation of data; in the writing of the manuscript, or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

FFNN	feed forward neural network
HML	Human Motion Laboratory
LS	least squares
Mocap	MOtion CAPture
MSE	Mean Square Error
NARX-NN	nonlinear autoregressive exogenous neural network
NN	neural network
PJAIT	Polish-Japanese Academy of Information Technology
RMSE	root mean squared error

1. Kitagawa, M.; Windsor, B. MoCap for artists: workflow and techniques for motion capture; Elsevier/Focal Press: Amsterdam ; Boston, 2008. OCLC: ocn190620556.
2. Menache, A. Understanding motion capture for computer animation, 2nd ed ed.; Morgan Kaufmann: Burlington, MA, 2011. OCLC: ocn641537758.
3. Mündermann, L.; Corazza, S.; Andriacchi, T.P. The evolution of methods for the capture of human movement leading to markerless motion capture for biomechanical applications. Journal of NeuroEngineering and Rehabilitation **2006**, 3, 6. doi:10.1186/1743-0003-3-6.
4. Windolf, M.; Götzen, N.; Morlock, M. Systematic accuracy and precision analysis of video motion capturing systems—exemplified on the Vicon-460 system. Journal of Biomechanics **2008**, 41, 2776–2780.
5. Yang, P.F.; Sanno, M.; Brüggemann, G.P.; Rittweger, J. Evaluation of the performance of a motion capture system for small displacement recording and a discussion for its application potential in bone deformation *in vivo* measurements. Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine **2012**, 226, 838–847. doi:10.1177/0954411912452994.
6. Jensenius, A.; Nymoen, K.; Skogstad, S.; Voldsund, A. A Study of the Noise-Level in Two Infrared Marker-Based Motion Capture Systems. Proceedings of the 9th Sound and Music Computing Conference, SMC 2012; , 2012; pp. 258–263.
7. Eichelberger, P.; Ferraro, M.; Minder, U.; Denton, T.; Blasimann, A.; Krause, F.; Baur, H. Analysis of accuracy in optical motion capture–A protocol for laboratory setup evaluation. Journal of Biomechanics **2016**, 49, 2085–2088.
8. Skurowski, P.; Pawlyta, M. On the Noise Complexity in an Optical Motion Capture Facility. Sensors **2019**, 19, 4435. Number: 20 Publisher: Multidisciplinary Digital Publishing Institute, doi:10.3390/s19204435.
9. Woltring, H.J. On optimal smoothing and derivative estimation from noisy displacement data in biomechanics. Human Movement Science **1985**, 4, 229–245. 00197, doi:10.1016/0167-9457(85)90004-1.
10. Giakas, G.; Baltzopoulos, V. A comparison of automatic filtering techniques applied to biomechanical walking data. Journal of Biomechanics **1997**, 30, 847–850. 00097, doi:10.1016/S0021-9290(97)00042-0.

11. Skurowski, P.; Pawlyta, M. Functional Body Mesh Representation,, A Simplified Kinematic Model, Its Inference and Applications. Applied Mathematics & Information Sciences **2016**, 10, 71–82. doi:10.18576/amis/100107.
12. Liu, G.; McMillan, L. Estimation of missing markers in human motion capture. The Visual Computer **2006**, 22, 721–728. doi:10.1007/s00371-006-0080-9.
13. Gløersen, Ø.; Federolf, P. Predicting Missing Marker Trajectories in Human Motion Data Using Marker Intercorrelations. PLOS ONE **2016**, 11, e0152616. Publisher: Public Library of Science, doi:10.1371/journal.pone.0152616.
14. Tits, M.; Tilmanne, J.; Dutoit, T. Robust and automatic motion-capture data recovery using soft skeleton constraints and model averaging. PLOS ONE **2018**, 13, e0199744. Publisher: Public Library of Science, doi:10.1371/journal.pone.0199744.
15. Camargo, J.; Ramanathan, A.; Csomay-Shanklin, N.; Young, A. Automated gap-filling for marker-based biomechanical motion capture data. Computer Methods in Biomechanics and Biomedical Engineering **2020**, 23, 1180–1189. Publisher: Taylor & Francis _eprint: https://doi.org/10.1080/10255842.2020.1789971, doi:10.1080/10255842.2020.1789971.
16. Kaufmann, M.; Aksan, E.; Song, J.; Pece, F.; Ziegler, R.; Hilliges, O. Convolutional Autoencoders for Human Motion Infilling. arXiv:2010.11531 [cs] **2020**. arXiv: 2010.11531.
17. Zhu, Y. Reconstruction of Missing Markers in Motion Capture Based on Deep Learning. 2020 IEEE 3rd International Conference on Information Systems and Computer Aided Education (ICISCAE), 2020, pp. 346–349. doi:10.1109/ICISCAE51034.2020.9236900.
18. Royo Sánchez, A.C.; Aguilar Martín, J.J.; Santolaria Mazo, J. Development of a new calibration procedure and its experimental validation applied to a human motion capture system. Journal of Biomechanical Engineering **2014**, 136, 124502. doi:10.1115/1.4028523.
19. Nagymáté, G.; Tuchband, T.; Kiss, R.M. A novel validation and calibration method for motion capture systems based on micro-triangulation. Journal of Biomechanics **2018**, 74, 16–22. doi:10.1016/j.jbiomech.2018.04.009.
20. Weber, M.; Amor, H.B.; Alexander, T. Identifying Motion Capture Tracking Markers with Self-Organizing Maps. 2008 IEEE Virtual Reality Conference, 2008, pp. 297–298. ISSN: 2375-5334, doi:10.1109/VR.2008.4480809.
21. Jiménez Bascones, J.L.; Graña, M.; Lopez-Guede, J.M. Robust labeling of human motion markers in the presence of occlusions. Neurocomputing **2019**, 353, 96–105. doi:10.1016/j.neucom.2018.05.132.
22. Ghorbani, S.; Etemad, A.; Troje, N.F. Auto-labelling of Markers in Optical Motion Capture by Permutation Learning. Advances in Computer Graphics; Gavrilova, M.; Chang, J.; Thalmann, N.M.; Hitzer, E.; Ishikawa, H., Eds.; Springer International Publishing: Cham, 2019; Lecture Notes in Computer Science, pp. 167–178. doi:10.1007/978-3-030-22514-8_14.
23. Han, S.; Liu, B.; Wang, R.; Ye, Y.; Twigg, C.D.; Kin, K. Online optical marker-based hand tracking with deep labels. ACM Transactions on Graphics **2018**, 37, 166:1–166:10. doi:10.1145/3197517.3201399.
24. Regression analysis - Encyclopedia of Mathematics.
25. Stapor, K. Introduction to Probabilistic and Statistical Methods with Examples in R; Intelligent Systems Reference Library, Springer International Publishing, 2020. doi:10.1007/978-3-030-45799-0.
26. Savitzky, A.; Golay, M.J.E. Smoothing and Differentiation of Data by Simplified Least Squares Procedures. Analytical Chemistry **1964**, 36, 1627–1639. doi:10.1021/ac60214a047.
27. Hornik, K. Approximation capabilities of multilayer feedforward networks. Neural Networks **1991**, 4, 251–257. doi:10.1016/0893-6080(91)90009-T.
28. Insua, D.R.; Müller, P. Feedforward Neural Networks for Nonparametric Regression. In Practical Nonparametric and Semiparametric Bayesian Statistics; Dey, D.; Müller, P.; Sinha, D., Eds.; Lecture Notes in Statistics, Springer: New York, NY, 1998; pp. 181–193. doi:10.1007/978-1-4612-1732-9_9.
29. Czekalski, P.; Łyp, K. Neural network structure optimization in pattern recognition. Studia Informatica **2014**, 35.
30. Pillai, I.; Fumera, G.; Roli, F. Designing multi-label classifiers that maximize F measures: State of the art. Pattern Recognition **2017**, 61, 394–404. doi:10.1016/j.patcog.2016.08.008.
31. Gorodkin, J. Comparing two K-category assignments by a K-category correlation coefficient. Computational Biology and Chemistry **2004**, 28, 367–374. doi:10.1016/j.compbiolchem.2004.09.006.