Feature Descriptors for Gait Analysis From Depth Sensors

August 19, 2015

Contents

1 Introduction

Gait analysis plays an important part in the treatment and assessment of a number of medical conditions. Presently gait analysis is usually performed through a combination of visual assessment by an experienced physiotherapist, automated methods such as marker based motion capture, pressure sensitive walkways or accelerometers. It requires patients to travel to a gait assessment laboratory which is far from ideal for patients who have difficulty walking.

This problem, and a range of other healthcare challenges, is being tackled through research and development by the SPHERE (a Sensor Platform for Healthcare in a Residential Environment) group in Bristol. An automatic, in home, gait analysis pipeline has been designed [**?**, **?**] which assesses the quality of a subjects movement using inexpensive RGB-D cameras such as the Microsoft Kinect.

Currently this system uses joint position information captured by the OpenNI skeleton tracking software, based on the algorithm of [**?**]. This skeleton tracking software infers the 3D coordinates of each of the body's relevant joints producing a $n_{joints} \times 3$ dimensional vector. This data is then processed using a manifold learning method, Diffusion maps [**?**], to reduce the dimensionality of the

data. This method builds up a 3D representation of the types of body configurations displayed in a training dataset containing footage of the motion being measured. New skeleton data is then projected onto this manifold. This effectively parameterises the motion, removing the redundant information contained in the skeleton data, and enabling simple comparison of poses. [1] Finally, a statistical model of normal gait is built up from the training data using these pose vectors. New data is compared with this model producing a quality score for both pose and dynamics on a frame-by-frame basis.

Since this system uses data driven, machine learning methods to learn both the manifold representation of pose and the model of normal motion, it can be applied to other types of movement quality assessment such a sports movement optimisation or physiotherapy exercise coaching. The system has been applied to a sitting-standing motion, to punching motions in boxing$^{howtositehttp://www.irc-sphere.ac.uk/work-package-2/movem}$ and to people walking upstairs.

One issue currently limiting the effectiveness of this system is the fragility of the skeleton tracking software. Shotton et al's algorithm was designed for controlling entertainment/gaming systems with the user viewed frontally, within a range of 1-4m and at a pitch angle near 0°. Outside of these conditions skeletons become noisy and unreliable. Typically only a small fraction of data recorded from a camera attached to the ceiling above the stairs is fit for use with the system. Increasing amount of usable data requires more intrusive camera placement which is to be avoided. The skeleton trackers also perform poorly when props are involved in the scene, for example grasping a banister or a ball often leads to erroneous joint positions for that arm. It also struggles to accurately record sitting/standing motions.

The aim of this project is to develop a tailor made system for determining the reduced pose vector directly from RGB-D footage. To enable the flexibility of the rest of the system we require this new component to exhibit the same flexibility as the rest of the system by being able to record a wide range of motions. It should also work with an effective accuracy under the kinds of viewing angles produced by practical, unobtrusive, in home camera placements. This requires a data driven approach since the pose representation we wish to infer is not fixed, differing based on the body configurations presented in training data.

The methodology we find most suited to this task is a convolutional neural network (CNN). CNN's are a supervised learning method for extracting features, e.g. the pose vector, from images. Given training images labelled with the expected output the network extracts progressively higher level features representations leading to the final pose vector. Following training the network is then able to generalise to unseen data, producing an output inferred from the examples it has seen.

CNNs have been effectively applied to 2D human pose estimation from RGB images [**?**,**?**,**?**,**?**,**?**,**?**,**?**] where the positions of joints in the image plane were inferred. In [**?**] they were also applied to 3D joint position estimation from RGB, where they were shown to have reasonable accuracy from a range of viewing angles when trained with data captured by 4 cameras placed around the subjects. They have also been shown to benefit from depth depth data in the tasks of object detection [**?**], object pose estimation [**?**] and object recognition [**?**].

For assessing the effectiveness of our solution we focus on the staircase ascent motion, as this is the motion for which we possess the largest dataset. Refered to as the SPHERE staircase 2014 dataset [**?**], this includes 48 sequences of 12 individuals walking up stairs, captured by a Asus Xtion depth sensor

[1] We will refer to the projected points in this space as the pose vector, and to the skeleton data as the body configuration or joint position vector.

placed at the top of the stairs in a frontal and downward-looking position. It contains three types of abnormal gaits with lower-extremity musculoskeletal conditions, including freezing of gait and using a leading leg, left or right, in going up the stairs. All frames have been manually labelled as normal or abnormal by a qualified physiotherapist. There are 17 sequences of normal walking from 6 individuals and 31 sequences from the remaining 6 subjects with both normal and abnormal walking.

The accuracy of our predicted pose vectors are measured by computing the mean squared error (MSE) of the produced pose vectors against the label values. We also measure the change in overall system performance (how well the measured gait quality score matches the score labelled by a trained physiotherapist).

To the best of our knowledge this project will be the first time that CNNs will be applied to a 3D human pose estimation task on RGB-D images. It will also be a novel combination of CNNs and manifold learning methods. We find that this makes the CNN easier to train and more effective overall since it has far less outputs to specify. If this is proved to be the case it could potentially be applied to other tasks that have been attempted with CNNs such as human action recognition.

2 Background and Related Work

2.1 Depth Imaging

In depth images each pixel value represents the distance of that point from the camera. Depth images are unaffected by changes in lighting or human appearance. They provide a 3D map of the scene making background-foreground separation far easier which can often simplify computer vision tasks. [**?**]

There are three main technologies used to produce depth images: Time of flight (ToF) cameras, Stereo imaging cameras and Structured light cameras. It was not until the last 5 years that affordable good quality RGB-D sensors came on the market, since then there has been an explosion in their use in the computer vision community [**?**].

The data in the SPHERE staircase dataset was captured using a Asus Xtion Pro Live which uses the structured light technology developed by Primesense (same as the Microsoft Kinect). It consists of an infrared laser emitter, an infrared camera, which together make up the depth sensor, and an RGB camera. An infrared laser is passed through a diffraction grating to produce a known pattern of dots that is projected onto the scene then reflected back and captured by the infrared camera. The measured pattern is compared to a reference pattern produced at a known distance of reflection, which has been stored during the calibration process. The surface of reflection being farther or nearer than the reference surface produces a shift in the pattern which is used to determine the depth value [**?**, **?**].

2.1.1 Sensor Performance

Most of the studies reported below have focused on the Microsoft Kinect however both sensor contain the same depth sensing system and have been shown to perform equivalently when compared [**?**], hence we report the findings based off of the Kinect performance.

The range of the depth sensor is 0.8-3.5m with increasingly noisy or incomplete readings up to 8m. It has a 43° vertical by 57° horizontal field of view [**?**].

Stoyanov et al. [?] compare the performance of the Kinect with that of two other ToF depth imaging cameras (SwissRanger SR-4000 and Fotonic B70 ToF) assessing them against a ground truth of expensive and low fps laser depth scanner measurements. They find that within a range of 3.5m the Kinect outperforms that of the ToF sensors and is comparable to the laser scanner, and that outside of this range the accuracy falls considerably.

Both Khoshelham & Elberink [?] and Smisek et al. [?] have measured this effect experimentally comparing Kinect measurements with those from high performance laser scanners. They find temporally fluctuating noise in the depth measurements increases quadratically with distance from the sensor so the depth precision decreases from about 0.5cm at 1m to 7cm at 7m. Nguyen et al. shows there to also be linearly increasing noise with lateral distance, and greatly increased noise on surfaces at greater than 70° angles [?]. This last effect can lead to increased levels of noise around edges of humans.

As well as noise, the structured light sensors often return 'unknown' depth value pixels, known as holes, when the infrared receiver cannot read the reflected pattern properly. This can occur around the sides of foreground objects due to the slightly different viewing angles between the projector and camera as in regions 2 and 3 of figure **??**, or when certain surface materials, such as human hair, interfere with the infrared pattern's reflection as in region 4 in figure **??**.

It should be noted that each of these studies mentioned above [?, ?, ?] fail to consider the environmental factors in the quality of the measurement. Fiedler & Muller [?] show that air draft can cause changes of the depth values up to 21mm at a distance of 1.5m, and temperature variations cause changes up to 1.88mm per 1° C. They also find a temperature dependant drift in the position of objects captured by the RGB camera.

2.2 Human Pose Estimation

Human pose estimation (HPE) is generally considered the task of measuring in 2D or 3D the joint positions of the human body. It is one of the most researched problems in computer vision due to the difficultly of the problem and the variety of applications such as video surveillance, humancomputer interaction, digital entertainment and sport science as well as medical applications.

This is a difficult task for a number of reasons. Firstly the human body has around 20 degrees of freedom [?], producing a huge space of possible body configurations, many of which will cause some joints to be occluded when viewed from a single camera. Additional difficulties arise from the variety in human appearance and clothing, and from left right ambiguities. Traditional motion capture methods (MoCap) methods rely on markers attached to the subject and multiple cameras to overcome these issues. Whilst such systems can provide highly accurate pose data, their use is restricted to controlled environments using expensive and calibrated recording equipment which renders them unsuitable in many applications.

Monocular visual pose estimation methods (reviewed in [?, ?, ?, ?, ?]) are generally divided into two approaches (e.g. by [?]); model based (or generative) and model-free (or discriminative) approaches. Model based approaches use prior knowledge of human shape and kinematics such as fixed limb lengths and defined joint angle limits to cast the image to pose transformation as a nonlinear optimisation problem or probabilistically in terms of a likelihood function, i.e. given this image data (and sometimes previous frames pose knowledge) what is the most likely valid pose. Model-free approaches instead learn a direct mapping from image data to pose, generally requiring learning/example based methods

to achieve this. Some 'hybrid' approaches combine the two using model-free methods as an initialiser to model based methods.

2.2.1 Human Pose Estimation Using Dimensionality Reduction

With both of the above approaches there are significant issues posed by the high dimensionality of pose data. In model based approaches likelihood functions, which are usually multi-modal and non-Gaussian, require a randomised search [?]. Such searches in 20 dimensions are computationally expensive and often lead to super real time frame rates [?]. In model free approaches training data must account for the highly non-linear mapping between image and pose, which means that the pose space must be densely sampled in the training set. Densely sampling a 20 dimensional space, even only the parts that correspond to valid human motion, whilst also modelling all the invariant aspects such as body shapes, viewing angle etc requires an inordinate amount of data [?, ?].

Although the full pose space is very large and high dimensional it has been shown e.g. in [?, ?] that if considering only the movements in a well defined activity e.g. walking then the pose data can be well represented by a low dimensional latent manifold. In a work closely related to our own Elgammal et al. [?] use a Local Linear Embedding method to generate a 1D manifold representation (embedded within a 3D space)$^{(FIGURE?)}$ of a walking motion from single sequences of silhouette images. They use a Generalised Radial Basis Function interpolation framework [?] (a form of neural network) to learn the nonlinear mappings from the manifold to silhouette image space and from the manifold to full 3D joint positions. They then invert these mappings to extract points on the manifold from silhouettes and 3D joint positions from the points on the manifold. In contrast, our work builds the manifold representation from 3D joint position data, this has the benefit that the same manifold representation is not tied to subject appearance, generalising naturally to multiple subjects, which is not the case in [?] (although they do introduce an solution for this problem in [?]). It is also unlikely that this method could be used to capture abnormality in gait since defining an image to manifold transformation explicitly from the inverse constrains all input images to the poses contained in the original sequence. Elgammal et al. argue that learning a smooth mapping from examples is an ill-posed problem unless the mapping is constrained since the mapping will be undefined in other parts of the space. However we show that having unseen (in the training set) poses mapped to points away from the manifold, which is precisely what an undefined mapping causes, is the a good indication of ab ^{Sortthislater}

Brand [?], also inferred 3D pose from silhouettes using an intermediate manifold representation. He uses a maximum a posteriori estimation for mapping between the image and manifold space. This uses information across the whole input sequence to find the most likely and consistent solutions in order resolve the ambiguities in the many to many silhouette to pose mapping. A solution of this form is unacceptable in our case as on of the key features of the SPHERE system is online measurement.

Similarly Urtasan et al. [?] used Scaled Gaussian Process Latent Variable Models (SGPLVM) [?] on single training sequences (walking and a golf swing) labelled with 2D joint positions. These models build a low dimensional manifold, and simultaneously, a continuous mapping between this and the 2D joint positions. They use the low dimensional representation to facilitate efficient maximum a posteriori based tracking in this space. Again, this is only suitable as an offline solution.

Tangkuampien and Suter [?] used Kernel Principle Component Analysis (KPCA) to learn low dimensional representations of 3D joint positions, and separately using the same method, a low

dimensional representation of silhouette images. They then learn a mapping between these spaces using Locally Linear Embedding (LLE) reconstruction. This has some similarity with using a CNN, where the image is transformed in to progressively more concise feature representations, before the regression takes place. The disadvantage of using LLE to perform the final transformation is that it is contained to within the manifold space contained in pose training data. As discussed above, our method has the benefit that it naturally maps unseen poses away from the manifold of the training data. This reduces the number of poses we must capture in our training data.

Rosales et al [**?**, **?**] used 3D joint position data from a MoCap system to render synthetic training data from multiple angles. Hu moments were used to extract visual features from these(and real) images. Unsupervised learning was used to cluster 3D joint position data into areas of similar pose and a neural network was trained separately on each cluster to learn the mapping from visual features to pose. Using the developments in training of deep neural networks since this work we show that it is feasible to have a single CNN both learn the features best suited and the mapping to all poses. This removes the need for clustering and separate networks leading to a simpler easily adaptable solution. Although their use of MoCap data as opposed to Kinect Skeletons for ground truth is a change we could expect to improve performance in our system. Similarly rendering synthetic training data from multiple angles would be a smart way of improving our viewing angle tolerance.

2.2.2 Human Pose Estimation From Depth Images

With the advent low cost commodity depth sensors challenging aspects of RGB HPE such as the variability in human appearance and scene lighting were greatly simplified. RGB-D also provides richer data for inferring 3D structure; human poses which could be appear identical when projected onto the 2D image plane can be distinguished. Full body HPE methods from single depth images are review in [**?**]. With the Kinect sensor and its bundled software packages (Kinect SDK or alternatively the open source OpenNI) low cost, flexible and reasonably accurate HPE is now available and has been employed in a huge variety of scientific applications [**?**, **?**].

The Kinect SDK and OpenNI skeleton trackers apply some inter-frame tracking algorithms to the single frame pose measurements of [**?**]. In this work Shotton et al. leveraged a large mocap 3D joint position dataset which they re-targetted onto a variety of synthetic body models before rendering as if captured from a Kinect, simulating sensor noise, camera pose, and crop position. Producing synthetic depth images data is far simpler than in RGB since depth is far more invariant to subject clothing and appearance changes. Using these generated depth images and a ground truth labelling of each pixel as one of 31 body parts they trained a randomised regression forrest to perform this body part classification at each pixel using simple and computationally efficient pixel wise features. They then use these classifications to infer actual joint position through a simple averaging and mean shift procedure. The whole algorithm operates in real time on the computational resources allowed to them on the Xbox gaming consoles GPU. [**?**] adapts this work by allowing pixel classifications of a number of surrounding joints to be used when estimating the joint position, rather than the single corresponding body part pixels as in [**?**]. This is shown to improve the quality of the prediction for occluded joints. A similar use of mocap joint position data for rendering synthetic images from multiple views is suggested as an ideal way for increasing the view angle and subject invariance of our system.

Another discriminative method is [**?**] where they find geodesic extrema (which are expected to

correspond to the feet, hands and head) from Dijkstras algorithm on a graph produced by connecting all depth pixels in the image into a map. These points are identified (as hands feet or head) by applying local shape descriptors around the area.

Other methods e.g. [**?**, **?**, **?**, **?**] have focused on improving temporal smoothness of the measured pose by combining such discriminative methods with model based temporal model based tracking methods.

In a recent attempt Chan et al [**?**] use 3D point cloud information and propose a viewpoint and shape histogram feature based off these point clouds. This feature is then used to categorise the pose based on the action being performed using an introduced action mixed model. Each action is prescribed its own low dimensional manifold which allows a human pose database containing a limited amount of data to probabilistically infer the full pose.

2.3 Convolutional Neural Networks

Convolutional Neural Networks (CNNs) are biologically inspired supervised learning systems for extracting features from images. Essentially their goal is to learn an approximation of the function

$$\boldsymbol{Y} = F(\boldsymbol{Z}, \boldsymbol{W}) \tag{1}$$

where $\boldsymbol{Z}$ is an input image, the output $\boldsymbol{Y}$ is the inferred pose vector for this image, and $\boldsymbol{W}$ are the trainable parameters of the network.

They consist of a set of filter maps, essentially matrices (or tensors if applying to multiple channels), which are applied repeatedly across the whole of the input. Each application of these filters produces an activation value which is the sum of each filter element multiplied by its corresponding input pixel value plus a shift term known as a bias. The filter is applied iteratively across the whole image, typically with some overlap. This builds up an 'activation map' for that filter. With such a layer being comprised of a number of filters. The activations of each filter are stacked together, becoming the new 'image' which is passed onto the next layer. This process is illustrated in figure 1. This type of layer, known as Convolution layers, are typically followed by a non-linear function which is what enables the CNN to learn non-linear transformations such as the image to pose transformation we require. Although it is possible to stack convolutional layers on top of each other they are often followed by a pooling layer (also called subsampling). The idea of pooling is to reduce the spatial size from the previous layer as seen in figure **??**. Operating on each depth slice individually, i.e. each filters activation, the pooling window moves across the image taking the values of the elements in the input, conglomerating them using some operation, typically taking the max value as seen in the right hand side of figure **??**. CNNs also typically contain one or more fully connected layers at the end. Being fully connected means that rather than having a small filter applied repeatedly across the input, a number of filters of the same dimensions as the input are applied to the whole volume. This then outputs a $1 \times 1 \times K$ volume where K is the number of filters. This is then identical to regular neural networks where each unit in a layer is connected to every unit in the next. In our case, fully connected layers take the final high level feature representation from the rest of the network and perform the final regression.

CNNs are trained using backpropagation and gradient descent. During training an error function is defined which quantifies the difference between the networks output and the desired output. In

[2] http://cs.stanford.edu/people/karpathy/convnetjs/

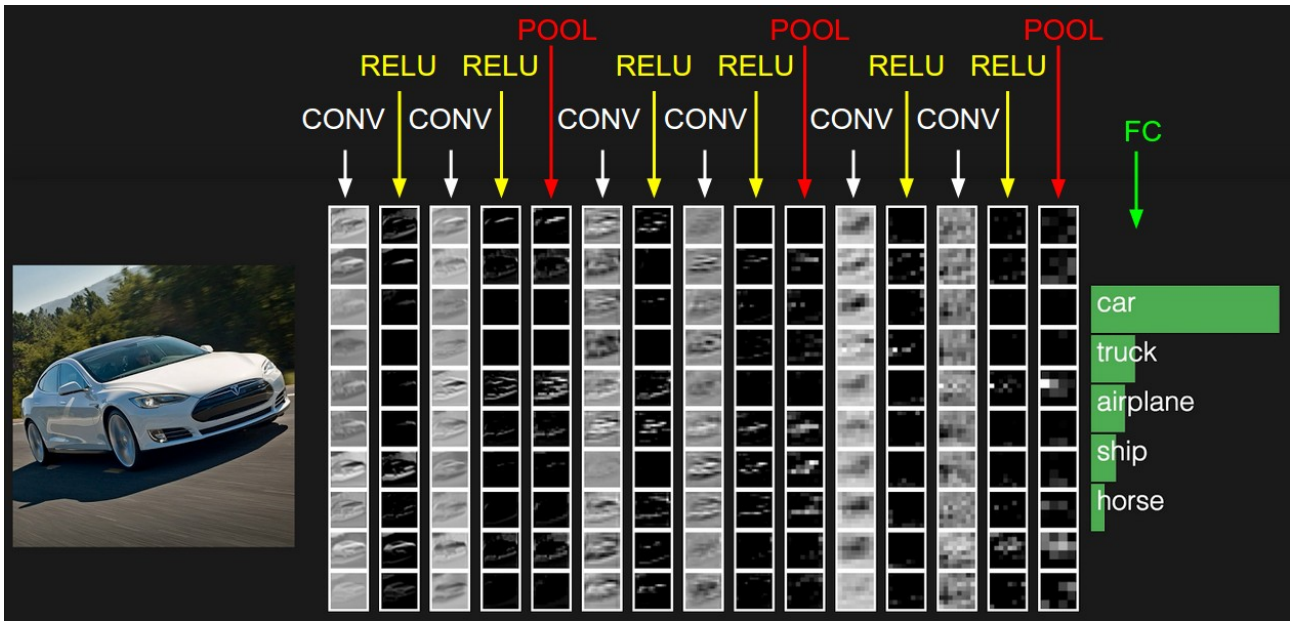

Figure 1: Shows a representation of the activations produced following a number of convolution layers, non-linearities (Rectified Linear Units, or ReLUs, are the function $max(0, x)$ and pooling layers. The network, an online demo from [**?**], is classifying images from the CIFAR-10 dataset using the ConvnetJS library[2].)

our case, as is typical of regression tasks, we use the euclidean distance between the two. Using the backpropagation algorithm [**?**] the derivative of this error with respect to each parameter is found. Then the values of each parameter are adjusted a small amount in the direction which reduces the error. In this way, over many training examples, the network converges on a minima in the error surface across the space of all parameter values. A derivation of backpropagation and further technical discussion of CNNs in general can be found in the preceding work on this project [**?**].

An advantage of CNNs over traditional computer vision methods is that rather than prescribing a hand-engineered feature such as the depth disparity feature in [**?**] or the view point and shape feature of [**?**], the network is responsible for learning features its self based off the data. This presents a significant advantage in our application since features which might be good for measuring pose for one type of motion may not be useful for other motions.

2.3.1 Human Pose Estimation Using CNNs

3 System Overview

4 Preprocessing

All data used in this project SPHERE-staircase2014 dataset [**?**])

Appendices

A Pre-existing SPHERE System For Movement Quality Analysis

In this section we will cover the existing system (as originally described in [**?**] and [**?**]) highlighting the points to be considered for our work.

As mentioned in section 4 the aim of the system is to quantify the quality of movement. This is achieved through comparing the recorded motion to a taught reference model of perfect motion. Although it has been evaluated specifically for gait measurements, the system aims to be widely applicable. Provided with suitable training data demonstrating perfect motion it could be applied to physiotherapy exercises, or in a sports movement optimisation application with little adaptation. This has been demonstrated with the system being applied to boxing and sitting-standing motions presented on the SPHERE web page[3].

The gait analysis system has been trained using the SPHERE-staircase2014 dataset [**?**]. This dataset includes 48 sequences of 12 individuals walking up stairs, captured by a Kinect camera placed at the top of the stairs in a frontal and downward-looking position. It contains three types of abnormal gaits with lower-extremity musculoskeletal conditions, including freezing of gait and using a leading leg, left or right, in going up the stairs. All frames have been manually labelled as normal or abnormal by a qualified physiotherapist. 17 sequences of normal walking from 6 individuals were used for training purposes and 31 sequences from the remaining 6 subjects with both normal and abnormal walking were kept for testing of the system. An example with the skeleton overlaid is shown in **??**.

The system can be considered as a pipeline, as represented in **??**. Once deployed in real home environments , another software component being developed in SPHERE will be responsible for human detection and recognition, hence we can presume input depth images similar to those contained in the dataset, and also knowledge of the person. The skeleton tracking packages from section **??** extract the joint positions of the body in the image. The skeletons are averaged over a temporal window to smooth noise before being scaled, rotated and translated to normalise the pose (these steps will be detailed further in section^{insertreferencelater}).

Next the low-level feature extraction stage builds a feature vector out of the skeleton data to encode the pose. Tao et al. tests and compares a number of viable feature representations for the skeleton data. They find that using a vector of the each joint coordinate concatenated performs best overall [**?**].

This joint position vector is then processed using the non-linear dimensionality reduction method Diffusion Maps [**?**]. The stage is first trained offline, building the manifold representation from the training subset of the data. The characteristic of manifolds produced by Diffusion Maps is that the data retain the relative euclidean distances between point in the reduced space. This representation is then extended to new data in the testing phase by projecting new skeletons onto the existing manifold.

Next the reduced pose vectors $\mathbf{Y}$ of the training data is used to learn two probabilistic continuous HMM (Hidden Markov Model) models, one of instantaneous pose, and one of dynamics. The normality of a pose is then the likelihood of the of $\mathbf{Y}$ being described by the pose model. Similarly the dynamical model computes a quality based on the likelihood of $\mathbf{Y}$ given the proceeding frames and the model of normal dynamics. Thresholds on these two normality scores are used to classify each frame normal or

[3] www.irc-sphere.ac.uk/work-package-2/movement-quality

abnormal.

The system is 'online' in that it measures abnormality on a frame-by-frame basis, rather than processing a recorded sequence offline and measuring across the full sequence. This enables it produce more detailed data on the parts of the motion that are deviating from normality, which is a benefit.

The contents...