STAG: Smart Tools and Applications in Graphics (2017)
A. Giachetti, P. Pingi and F. Stanco (Editors)

A 3 Cent Recognizer: Simple and Effective Retrieval and Classification of Mid-air Gestures from Single 3D Traces

F. M. Caputo[1], P. Prebianca[1], A. Carcangiu[2], L.D. Spano[3], A. Giachetti[1]

[1]Department of Computer Science, University of Verona, Italy
[2]Department of Electrical and Electronic Engineering, University of Cagliari, Italy
[3]Department of Mathematics and Information Technology, University of Cagliari, Italy

Abstract

In this paper we present a simple 3D gesture recognizer based on trajectory matching, showing its good performances in classification and retrieval of command gestures based on single hand trajectories. We demonstrate that further simplifications in porting the classic "1 dollar" algorithm approach from the 2D to the 3D gesture recognition and retrieval problems can result in very high classification accuracy and retrieval scores even on datasets with a large number of different gestures executed by different users. Furthermore, recognition can be good even with heavily subsampled path traces and with incomplete gestures.

CCS Concepts

•***Human-centered computing*** → ***Gestural input;***

1. Introduction

Mid-air gesture recognition can have a relevant role in the design of a variety of user interfaces for different applications: from games, simulators and immersive visualization environments to medical visualization [JW14], remote control of TV sets [IIGI11], and sign language decoding [LO98].

Many gesture recognition methods are based on image processing, exploiting pattern recognition tools, e.g. Hidden Markov Models [WH99], Dynamic Time Warping (DTW), Time-Delay Neural Networks (TDNN) and Finite-State Machines (FTM) [MA07, RA15,CYL16a] to analyze time series of image features. However, low cost hand/finger trackers like the Leap Motion Controller or Intel RealSense can provide an application with a good pre-processed input, considering the rather good reconstruction of 3D trajectories of palm and fingers.

This enables the direct use of these trajectories as input for the recognizer, filtering out a large amount of redundant information provided by the sensors.

For typical gesture recognition tasks, in fact, hand/finger trajectories should be sufficient to work with easily distinguishable classes. The reduction of the recognition task to a geometrical problem on 3D gesture trajectory can be used to provide simple tools for the design of custom gestural interfaces without the need of a large amount of training data [GCC*16].

A similar approach is indeed quite popular to design 2D stroke recognizers, where simple methods to compare gesture trajectories have been successfully used to create interfaces and interface design tools.

In this paper, we show the feasibility of this approach also on 3D gesture recognition based on trajectories obtained by low-cost hand trackers. The focus is on simple gestures that can be used in control interface, starting initially on single trajectories.

Considering a gesture dictionary proposed in a recent contest [SWV*17] and a novel dataset with many potential "command" gestures that may be useful in immersive Virtual Reality environments, we show that simple gesture recognizers based on 3D trajectories can achieve very good recognition/retrieval performances.

Pre-segmented gestures can be actually recognized even from single 3D trajectories sampled in 10 points or less, and even applying the method on largely incomplete gestures without relying on complex pattern recognition tools, but using only path distance evaluation on a few examples.

The rest of the paper is organized as follows. In Section 2 we present the related work in literature, in Section 3 we describe our method (i.e. the "3 cent recognizer"), in Section 4 we describe the dataset used for our tests, in Section 5 we present the results of the tests and in Section 6 we discuss the results and future plans.

2. Related Work

A huge amount of work on gesture recognition can be found in literature, using different input data, gesture encoding/modelling and application domain. Generic surveys can be found, for example in [PSH97, RA15, CYL16b], dealing also with static gestures,

DOI: 10.2312/stag.20171221

taxonomies of tracking methods and problem settings, application domains.

Considering the specific tasks of dynamic gesture recognition on which we are focusing, there are still many interesting works, mostly using image sequence processing and solving both for segmentation and gesture characterization [CYL16b].

A smaller amount of papers address the problem of gesture recognition considering only or mainly the hand position or hand skeleton data as the recognizer input. However several reasons could suggest this approach, given the fact that cheap sensors can now provide the input data with a good reliability. The first consists in the fact that hand/finger trajectories surely include sufficient/redundant information for the recognition of gestures and this approach works quite well for 2D touch interfaces. The second is that the use of the trajectories of body keypoints, provided by depth sensors APIs, are widely used for "action" recognition with good success, using different encoding methods for the skeleton nodes' sequences [HRHZ17].

Also few datasets with recorded hands/finger trajectories for gestures dictionaries have been proposed.

One of these is related to the SHREC 2017 contest on Hand Gesture Recognition Using a Depth and Skeletal Dataset [SWV*17]. Here a dataset with multiple hand/finger keypoints tracked during gestures that can be recognized in "natural" interfaces is provided.

For simple gesture recognition/retrieval actually a single trajectory could be enough. Algorithms for 3D trajectories comparison applied to gesture recognition has been presented in [SL15]. In [YYL16] a method to parse trajectories to derive a gesture descriptor based on sequences of segmented primitives is presented and tested also on a dataset of Australian sign language gestures.

The 3 dollar recognizer [KR10] is an adaptation of the popular "1 dollar" algorithm family [WWL07, VAW12] to 3D gestures, where the original "1 dollar" method is based on a 4-step algorithm consisting of resampling, rotating, centering and scaling and finally classification.

This class of algorithms is popular for gestural interface design as it allows a quick prototyping of specific recognizers given a small set of example gestures without the use of complex classifiers libraries or large training sets.

However, both the 3 dollar and the following method proposed by the same authors, "Protractor 3D" [KR11] rely on quite strange normalization options for gestures probably derived by the adaptation of the original 2D methods and have been tested on small sets of gestures.

In our work we show that, keeping the idea of comparing sampled trajectories, but adopting different and even simpler processing options it is possible to greatly improve the recognizer performances, allowing very good retrieval and classification results on gesture dictionaries including interface-like commands defined by a single hand trajectory.

3. The Proposed "3 cent" Recognizer

Our gesture recognition procedure is based on simple trajectory matching. However, our approach is different from the 3 dollar or Protractor 3D methods. Both these methods rescale gestures on a fixed size bounding box after a rough preliminary rotation or directly after centering. However, it is clear that the gesture rescaling based on the bounding box can heavily distort gestures features, as it depends on the reference frame. This rescaling can be replaced with a simpler and more reasonable length based scaling, making gesture length equal to 1. Then, the 3 dollar method relies on a first gesture reorientation, rescaling and a further tricky realignment before comparison, while Protractor 3D on a more efficient Procrustes-like gesture alignment finding rotation minimizing sum of squared distances between points, that is the output distance.

However, we observe that in most 3D gesture recognition tasks (e.g., remote commands for TV, immersive VR (Virtual Reality) interfaces, and in general for gestural interaction), directionality of gestures matters, so that rotating the gesture would result in relevant information loss. Therefore, after length normalization, we compute path distance of couples of trajectories after centering them in the centroid without rotating.

The resulting algorithm is quite simple, but, as shown in tests, also powerful. We call it the 3 cent recognizer, as it is actually even simpler than the "3 dollar" recognizer, albeit more effective. The processing step for the input gesture trajectories, typically acquired as 3D positions sampled uniformly in time, are the following:

- Resample the example and test acquired gestures with cubic spline interpolation as sequences of N equally spaced points.
- Scale gestures to a standard unit length
- Translate the gestures in order to put the centroid in the origin.
- Estimate the gestures' distance as sum of squared distance of corresponding points.

We assume that gestures are already segmented. However, as the method is quite efficient, it could be possible to use it to locate and recognize gestures in a raw and unsegmented stream of points. This result could be obtained by using simple heuristics for gesture segmentation or with a sliding window approach. These ideas, combined with the ability of our method to recognize gestures from a partial match (see Section 5.3), could result in a cheap and effective online recognizer. We plan to implement and test a similar solution in the near future.

4. Gestures Datasets

To compare the gesture recognizers, we used a novel dataset and exploited the dataset created for the "Shrec'17 contest on 3D Hand Gesture Recognition Using a Depth and Skeletal Dataset" [SWV*17].

4.1. 26 Gestures Datasets

The novel gesture dataset has been created to test the usability of the)3 cent recognizer on 26 different command interface gestures characterized by different 3D trajectories have been defined and performed by 14 different subjects using a Leap Motion. We release the dataset publicly on the web (https://github.com/davidespano/3cent-dataset). The dataset contains a sequence of 3D points, representing the position of the dominant-hand forefinger, together with the sample timestamp. During the ac-

quisition process, the user was instructed with an animation showing the ideal trajectory, which s/he had to replicate immediately afterwards.

The gesture set includes semi-circular arcs (arc3Dleft, arc3Dright), symbols drawn in a 3D space (caret, check, curly-bracket-left, curly-bracket-right, delete, pigtail, square-bracket-left, square-bracket-right, star, v, x), simple geometric figures (left-swipe, right-swipe, circle, rectangle, zig-zag), 3D polygonal chains (poly3Dxyz, poly3Dxzy, poly3Dyxz, poly3Dyzx, poly3Dzxy, poly3Dzyx) and a 3D spiral. Example gesture trajectories are shown in Fig. 1.

4.2. Shrec '17 Dataset

The dataset proposed for the Shrec'17 contest on 3D Hand Gesture recognition is also based on 3D hand trajectories, even if it includes both gesture characterized by single hand trajectories. The dataset is composed by acquisitions of complete hand skeleton movements for 14 gesture classes, 9 "coarse", e.g. not characterized by finger movements: tap (label 2), Swipe Right (7), Swipe Left (8), Swipe Up (9), Swipe Down (10), Swipe X (11), Swipe + (12), Swipe V (13), Shake (14), and 5 "fine", e.g. characterized by finger movements: Grab (1), Expand (3), Pinch (4), Rotation Clockwise (5), Rotation Counter Clockwise (6), where gesture should be characterized by relative finger movements. Example gesture trajectories are shown in Fig. 2.

5. Experimental Results

We performed gesture retrieval and classification tests on the novel 26-classes dataset and on the SHREC coarse (and full) datasets. We compared retrieval scores and classification accuracy obtained using the 3 cent path comparison. To understand the relative importance of the scaling and rotation effects in gesture distance effectiveness, we also tested an intermediate method using the length-based gesture scaling of the 3 cent method coupled followed by a Procrustes rotation done similarly to the Protractor 3D method.

5.1. Retrieval Scores on the New Interface Gesture Dataset

On the novel 26-gestures dataset we evaluated a set of retrieval scores used in most Eurographics SHREC contests, e.g. Nearest Neighbor (NN), First Tier (FT), Second Tier (ST), e-measure (E) and Discounted Cumulated Gain (DCG) [SMKF04]. Furthermore, Precision-Recall plots have been analyzed and from the PR curves the Mean Average Precision (MAP) (e.g. the average of all precision values computed for each subject in the retrieved list was estimated). Values reported in Table 1 and the Precision-Recall plots show the great improvement given by avoiding rotation and scaling to a reference trajectory length. 3 cent results are quite close to the perfect retrieval of all relevant gestures among the first 12 retrieved.

On the same dataset we tested a simple classification task similar to that reported in [KR11] and mimicking the typical recognizer system training expected for these kind of methods. We selected randomly 5 users as training set and used a simple K-Nearest Neighbor classification (we tested K=1,3,5) to assign label to the remaining "test" set including all the gesture of the remaining 8 subjects. Results are shown in Table 2. It is clear that the length scaling and translation without rotation provides the best results.

	NN	1-Tier	2-Tier	e	DCG	mAP
Protractor3D [KR11]	0.574	0.402	0.564	0.342	0.677	0.415
3 cent+rotation	0.633	0.483	0.669	0. 388	0.731	0.507
3 cent	0.965	0.825	0.917	0.514	0.949	0.844

Table 1: *Retrieval scores on the novel 26-gestures dataset.*

	Accuracy		
	1-NN	3-NN	5-NN
Protractor 3D	55.8	53.8	49.0
3 cent w rotation	61.1	62.0	59.1
3 cent	**96.9**	95.7	95.2

Table 2: *Classification accuracy with 3 cent, 3 cent with Procrustes rotation and Protractor 3D algorithm on the 26-classes dataset using 1-NN, 3-NN and 5-NN classifiers. Bold font indicates best result.*

The 3 cent results are nearly optimal in this case, while other path normalization/comparison methods are not so effective.

5.2. Shrec '17 Classification Task on Rough Gestures

In this case we performed two tests. First, we considered only the 9 classes of rough gestures of the dataset (labels 2, 6, 7, 8, 9, 10, 11, 12, 13, 14), and tried to characterize gesture using the palm trajectory only. With this choice, we setup a classification test using the training and test data split provided by the contest organization.

The confusion matrix 5 shows that errors are related to a couple of classes, "tap" and "swipe down", while others are recognized with very good accuracy. This is reasonable as the two gestures are quite similar for the palm movement and differ for finger actions.

If we test the NN classification on the whole dataset, including gestures that should not be discriminated by palm trajectory only, results are still reasonable (Table 4).

Looking at the confusion matrix of this test (Fig. 5) it is possible to see that the palm trajectory is not effective in "fine gesture" classes 3-6 as expected, it is still accurate for "rough classes" 7-14. Classes 1 (grab) and 2 (tap) seem actually incorrectly categorized, as grab, even if classified as "fine" is recognized well, while 2, considered "rough" is not well recognized looking at palm trajectory

	Accuracy		
	1-NN	3-NN	5-NN
Protractor 3D	39.8	37.7	38.1
3 cent w rotation	76.6	77.6	79.2
3 cent	90.2	91.5	**92.0**

Table 3: *Classification accuracy with 3 cent, 3 cent with Procrustes rotation and Protractor 3D algorithm on the 9-classes "coarse" SHREC dataset using 1-NN, 3-NN and 5-NN classifiers. Bold font indicates best result.*

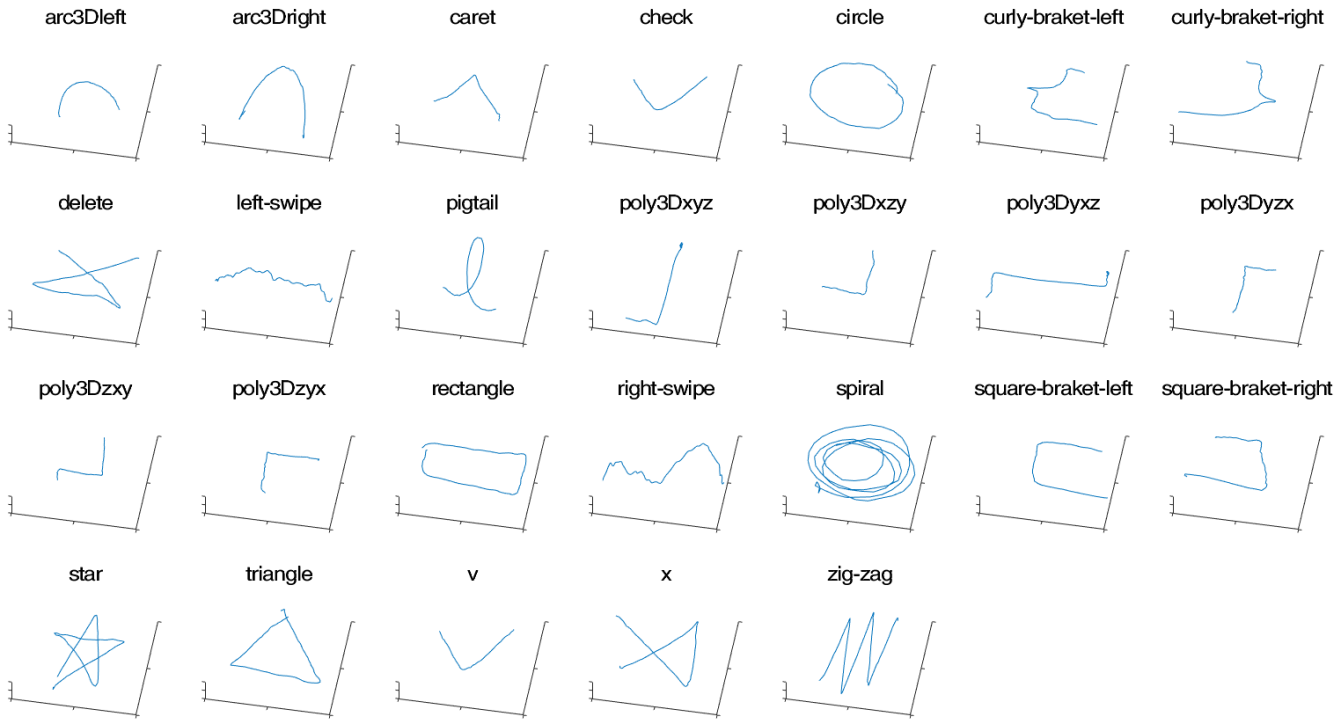

Figure 1: *Example trajectories for the 26 different gestures.*

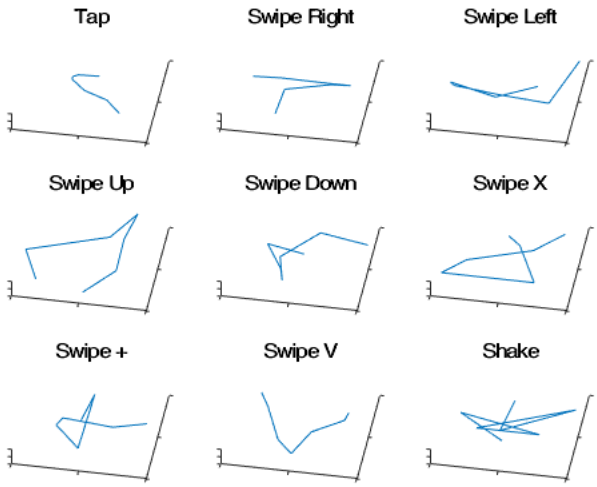

Figure 2: *Example palm trajectories for the SHREC "coarse" gesture classes.*

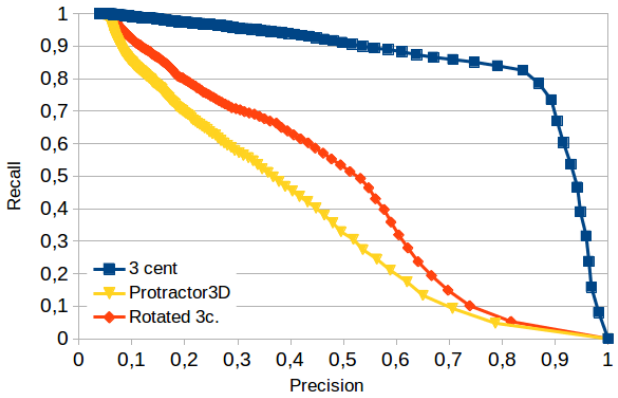

Figure 3: *Precision vs. Recall plot for the gesture retrieval task on the 26-gestures dataset.*

only. Looking at the single classes' accuracies, it is also possible to note that the method outperforms the best method in the contest for 5 gesture classes (swipe right, swipe left, swipe x, swipe +, shake).

"Fine" gestures are better recognized also just using the index tip trajectory instead of the palm, as shown in Table 5. The average accuracy obtained (77.9%) is actually not too far from the best method in the contest obtained with a Fisher Vector encoding of features coming from all the fingers' data (88.2%). We plan to exploit this fact to develop more advanced recognizer able to discriminate "coarse" and "fine" gesture and apply then specialized recognizers to the subclasses.

5.3. Subsampling and Recognition of Incomplete Gestures

As shown in the 2D case [Vat11], the dollar recognizer family is rather robust against point subsampling. Figs. 7 and 8 show that the

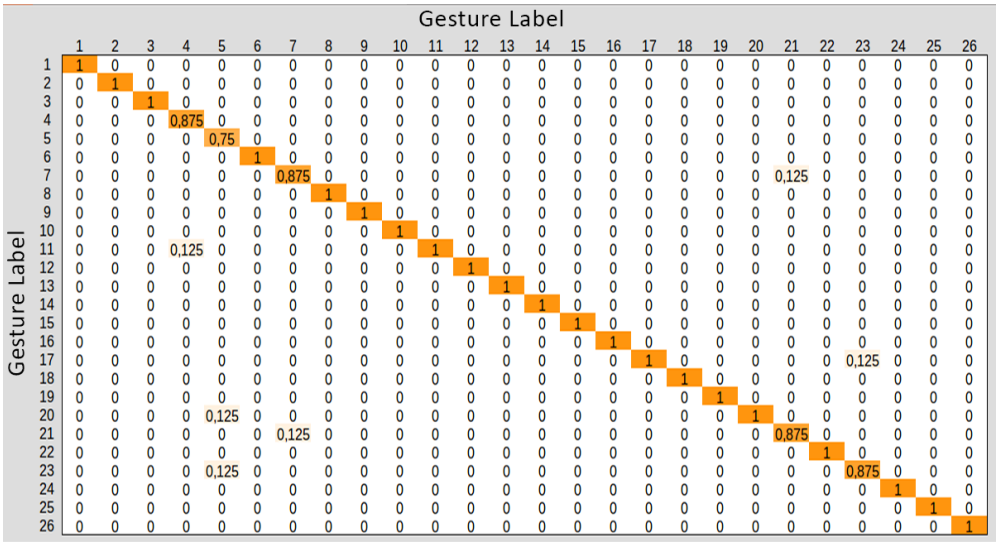

Figure 4: *Confusion matrix (percentage of assigned label for each test label) for the 26-classes dataset using 3 cent/Nearest Neighbor classification.*

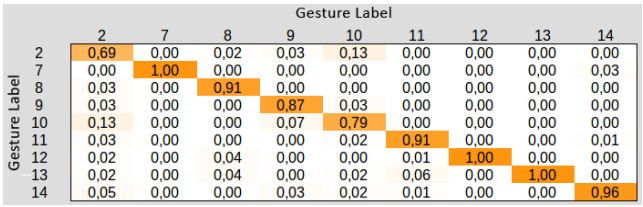

Figure 5: *Confusion matrix (percentage of assigned label for each test label) for the 9-classes "coarse" SHREC dataset using 3 cent Nearest Neighbor classification.*

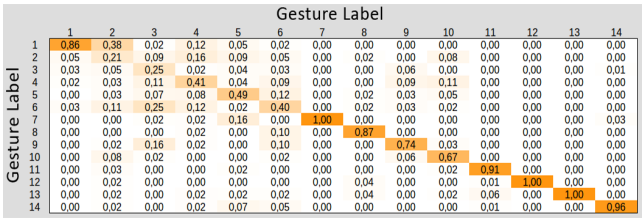

Figure 6: *Confusion matrix for the 14-classes SHREC dataset using 3 cent Nearest Neighbor classification.*

	Accuracy		
	1-NN	3-NN	5-NN
Protractor 3D	31.3	30.0	31.3
3 cent w rotation	56.4	56.7	60.6
3 cent	71.0	71.0	**71.9**

Table 4: *Classification accuracy with 3 cent, 3 cent with Procrustes rotation and Protractor 3D algorithms applied on the palm trajectory only on the 14-classes SHREC dataset (including also finger gestures) using 1-NN, 3-NN and 5-NN classifiers. Bold font indicates best result.*

	Accuracy		
	1-NN	3-NN	5-NN
Protractor 3D	41.4	41.2	39.6
3 cent w rotation	49.9	49.6	50.1
3 cent	75.7	77.1	**77.9**

Table 5: *Classification accuracy with 3 cent, 3 cent with Procrustes rotation and Protractor 3D algorithms applied on the index tip trajectory only on the 14-classes SHREC dataset (including also finger gestures) using 1-NN, 3-NN and 5-NN classifiers. Bold font indicates best result.*

classification accuracies on the 26-gestures and on the 9-gestures reduced SHREC tests are practically unchanged reducing the sampling up to 10 equally spaced points and are still reasonable even using only three points.

An even more interesting outcome is that the classification accuracy also stays practically unchanged if the match is done on the first 60% of the gesture length (see Figs. 7 and 8). This is a really interesting result, showing that even on datasets with many template classes, recognition can be performed before the gesture completion.

This fact and the efficiency of the method, suggest also the possibility of developing effective online gesture recognizers based on simple trajectories comparisons as well as the design of visual feedback mechanisms suggesting gesture completion from already performed trajectories as suggested in [GCC*16].

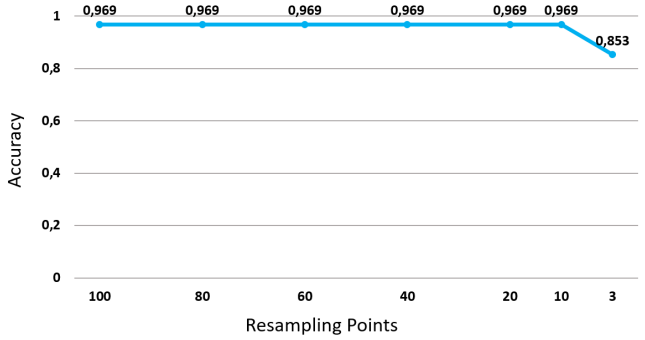

Figure 7: *Effect of changes in trajectory sampling. Accuracy in NN classification on the 26 gestures database is practically unchanged using a sampling with just 10 points.*

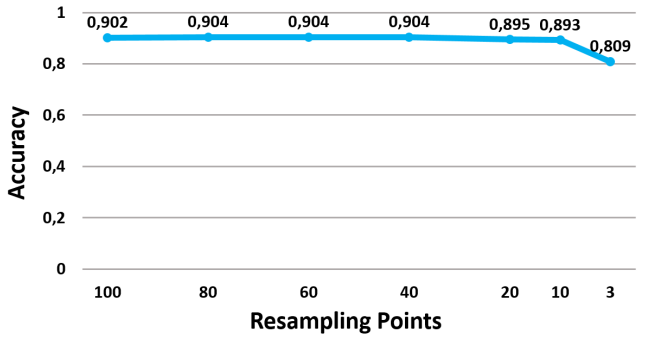

Figure 8: *Effect of changes in trajectory sampling. Accuracy in 3 cent/NN classification on the 9 "coarse" gestures of the SHREC database is practically unchanged using a sampling with just 10 points.*

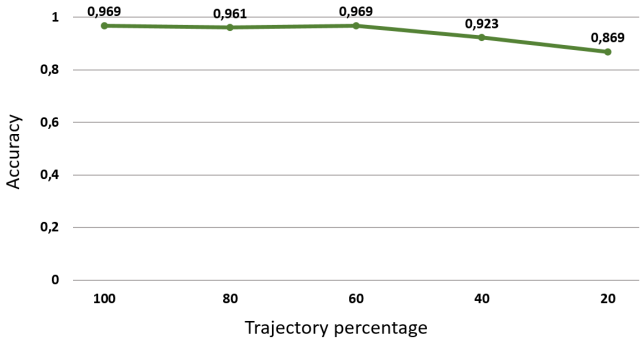

Figure 9: *Accuracy in 3 cent/NN classification on the 26 gestures database is practically unchanged if gestures are truncated at* 60% *of the trajectory length and is still good after just* 20%.

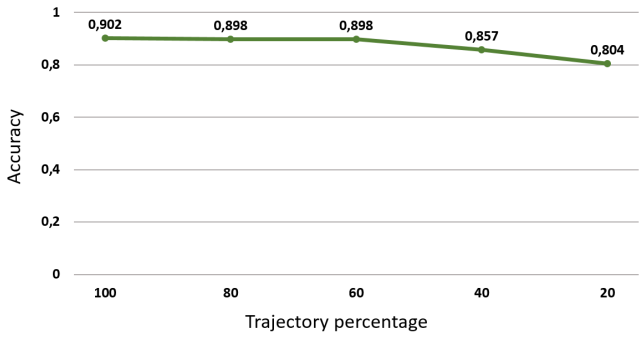

Figure 10: *Accuracy in 3 cent/NN classification on the 9 "coarse" gestures SHREC database is practically unchanged if gestures are truncated at* 60% *of the trajectory length and is still good after just* 20%.

6. Discussion

Simple gesture recognizer such as those of the "dollar" family, have been proposed for the use in touchscreen interaction due to their simplicity and effectiveness, that gives the possibility of prototyping novel gestural interface without the need of complex training.

In this paper we have shown that is approach is feasible also in 3D if we drop normalization methods derived from the 2D case and we adopt even simpler solutions that take into account the characteristics of typical mid-air gestures designed for user interaction, obtaining very good classification performances on gestures characterized by single hand trajectories.

We plan to test this approach also on datasets related to other tasks like sign language recognition.

A limitation of the method is related to the fact that it assumes that each gesture is performed in a well defined way, with fixed direction and orientation, or that all the possible different realizations of gestures with the same label are represented in the example set. It has to be noted that other steps are necessary to obtain an efficient and reliable online gesture recognizer (e.g. segmentation or multiple windows comparison strategies for accurate gesture location in hand motion sequences), and the handling of multiple trajectories in order to classify gestures characterized not only by palm motion, but also by finger movements. Furthermore, it will be necessary to determine specific methods to avoid false detections in generic online sequence processing with specific heuristics.

We plan to focus on these issues as future work in order to build an effective 3D gestural interface designer tool based on simple path comparison based online recognizer.

References

[CYL16a] CHENG H., YANG L., LIU Z.: Survey on 3d hand gesture recognition. *IEEE Trans. Circuits Syst. Video Techn. 26*, 9 (2016), 1659–1673. doi:10.1109/TCSVT.2015.2469551. 1

[CYL16b] CHENG H., YANG L., LIU Z.: Survey on 3d hand gesture recognition. *IEEE Trans. Cir. and Sys. for Video Technol. 26*, 9 (Sept. 2016), 1659–1673. URL: https://doi.org/10.1109/TCSVT.2015.2469551, doi:10.1109/TCSVT.2015.2469551. 1, 2

[GCC*16] GIACHETTI A., CAPUTO F. M., CARCANGIU A., SCATENI R., SPANO L. D.: Shape Retrieval and 3D Gestural Interaction. In *Eurographics Workshop on 3D Object Retrieval* (2016), Ferreira A., Giachetti A., Giorgi D., (Eds.), The Eurographics Association. doi:10.2312/3dor.20161079. 1, 5

[HRHZ17] HAN F., REILY B., HOFF W., ZHANG H.: Space-time representation of people based on 3d skeletal data: A review. *Computer Vision and Image Understanding 158* (2017), 85–105. 2

[IIGI11] IONESCU D., IONESCU B., GADEA C., ISLAM S.: An intelligent gesture interface for controlling tv sets and set-top boxes. In *Applied Computational Intelligence and Informatics (SACI), 2011 6th IEEE International Symposium on* (2011), IEEE, pp. 159–164. 1

[JW14] JACOB M. G., WACHS J. P.: Context-based hand gesture recognition for the operating room. *Pattern Recognition Letters 36* (2014), 196–203. 1

[KR10] KRATZ S., ROHS M.: A $ 3 gesture recognizer: simple gesture recognition for devices equipped with 3d acceleration sensors. In *Proceedings of the 15th international conference on Intelligent user interfaces* (2010), ACM, pp. 341–344. 2

[KR11] KRATZ S., ROHS M.: Protractor3d: a closed-form solution to rotation-invariant 3d gestures. In *Proceedings of the 16th international conference on Intelligent user interfaces* (2011), ACM, pp. 371–374. 2, 3

[LO98] LIANG R.-H., OUHYOUNG M.: A real-time continuous gesture recognition system for sign language. In *Automatic Face and Gesture Recognition, 1998. Proceedings. Third IEEE International Conference on* (1998), IEEE, pp. 558–567. 1

[MA07] MITRA S., ACHARYA T.: Gesture recognition: A survey. *IEEE Trans. Systems, Man, and Cybernetics, Part C 37*, 3 (2007), 311–324. doi:10.1109/TSMCC.2007.893280. 1

[PSH97] PAVLOVIC V. I., SHARMA R., HUANG T. S.: Visual interpretation of hand gestures for human-computer interaction: A review. *IEEE Transactions on pattern analysis and machine intelligence 19*, 7 (1997), 677–695. 1

[RA15] RAUTARAY S. S., AGRAWAL A.: Vision based hand gesture recognition for human computer interaction: a survey. *Artif. Intell. Rev. 43*, 1 (2015), 1–54. doi:10.1007/s10462-012-9356-9. 1

[SL15] SHAO Z., LI Y.: Integral invariants for space motion trajectory matching and recognition. *Pattern Recognition 48*, 8 (2015), 2418–2432. 2

[SMKF04] SHILANE P., MIN P., KAZHDAN M., FUNKHOUSER T.: The princeton shape benchmark. In *Shape modeling applications, 2004. Proceedings* (2004), IEEE, pp. 167–178. 3

[SWV*17] SMEDT Q. D., WANNOUS H., VANDEBORRE J.-P., GUERRY J., SAUX B. L., FILLIAT D.: 3D Hand Gesture Recognition Using a Depth and Skeletal Dataset. In *Eurographics Workshop on 3D Object Retrieval* (2017), Pratikakis I., Dupont F., Ovsjanikov M., (Eds.), The Eurographics Association. doi:10.2312/3dor.20171049. 1, 2

[Vat11] VATAVU R.-D.: The effect of sampling rate on the performance of template-based gesture recognizers. In *Proceedings of the 13th international conference on multimodal interfaces* (2011), ACM, pp. 271–278. 4

[VAW12] VATAVU R.-D., ANTHONY L., WOBBROCK J. O.: Gestures as point clouds: a $ p recognizer for user interface prototypes. In *Proceedings of the 14th ACM international conference on Multimodal interaction* (2012), ACM, pp. 273–280. 2

[WH99] WU Y., HUANG T. S.: Vision-based gesture recognition: A review. In *Gesture Workshop* (1999), vol. 1739, Springer, pp. 103–115. 1

[WWL07] WOBBROCK J. O., WILSON A. D., LI Y.: Gestures without libraries, toolkits or training: a $ 1 recognizer for user interface prototypes. In *Proceedings of the 20th annual ACM symposium on User interface software and technology* (2007), ACM, pp. 159–168. 2

[YYL16] YANG J., YUAN J., LI Y.: Parsing 3d motion trajectory for gesture recognition. *Journal of Visual Communication and Image Representation 38* (2016), 627–640. 2