Towards Haptic Performance Analysis Using K-Metrics

Richard Hall[1], Hemang Rathod[1], Mauro Maiorca[1], Ioanna Ioannou[1], Edmund Kazmierczak[2], Stephen O Leary[3], and Peter Harris[4]

[1] Melbourne University Virtual Environment for Simulation
[2] Department of Computer Science and Software Engineering
[3] Department of Otolaryngology
[4] Biomedical Multimedia Unit, Faculty of Medicine, Dentistry and Health Sciences, University of Melbourne, Victoria, Australia

Abstract. It is desirable to automatically classify data samples for the assessment of quantitative performance of users of haptic devices as the haptic data volume may be much higher than is feasible to manually annotate. In this paper we compare the use of three k-metrics for automated classifaction of human motion: cosine, extrinsic curvature and symmetric centroid deviation. Such classification algorithms make predictions about data attributes, whose quality we assess via three mathematical methods of comparison: root mean square deviation, sensitivity error and entropy correlation coefficient. Our assessment suggests that k-cosine might be more promising at analysing haptic motion than our two other metrics.

Keywords: Haptic performance analysis, motion classification.

1 Introduction

The number of devices that interface between user and computer using different sensory modalities are continuously increasing as is their applications, including gaming entertainment, intelligent impairment assistance, and training. Haptics enabled virtual reality simulators are becoming a key adjunct to traditional methods of surgical training, providing trainee surgeons with the '*feel*' of a procedure as well as its visual aspects by combining visual, haptic, and auditory interfaces. Within our domain of interest there exist a number of visuo-haptic simulators for temporal bone surgery [1,2,3,4,5,6,7], which purport to imitate real-world interactions between surgical instruments and temporal bone anatomy whilst providing appropriate visuo-haptic feedback to surgical trainees.

Aside from developing simulations of surgical procedures, researchers have also developed ways to evaluate trainee performance on these simulations [8]. The simulators by Sewell, Morris et.al. [2,9] produce a number of metrics that facilitate evaluation of the trainee's ability to remove the correct amount of bone using the correct drilling tools and drill speed while drilling at a *safe* distance from sensitive anatomical structures. However, we know of no prior work in analysing haptic motion in the way we describe in this paper.

A. Pirhonen and S. Brewster (Eds.): HAID 2008, LNCS 5270, pp. 50–59, 2008.

In temporal bone surgery, the technique used to remove bone is to sweep the surgical drill across the bone in a motion that is often referred to as a *stroke*. The choice of stroke is critical to the success of a procedure. For example, an expert will shorten their strokes as they drill in proximity of a sensitive anatomical structure such as the facial nerve, and lengthen their strokes when they are certain that are in an area in which they can work quickly.

This papers evaluates three methods for the automatic annotation of streams of haptic data into strokes. It is organised as follows. In Section 2 we describe the factorial design for our experiment and in Section 3 we discuss the specific metrics that we use to automatically annotate strokes. In Section 4 we discuss several methods to assess the predictive performance of our algorithms. In Section 5 we apply the metrics to ten data, assess the metrics, and present the results of this comparison. Subsequently there is a brief discussion and conclusion.

2 Experimental Design

What exactly is a stroke? Our haptic tool constantly streams out positional information, which is essentially a time series T of three dimensional points P:

$$T = \{P_j = (x_j, y_j, z_j) | j = 1, 2, 3, \ldots, m\}; \tag{1}$$

where P_j denotes a *temporal* neighbour of P_{j-1} and P_{j+1}, and (x_j, y_j, z_j) is the Cartesian coordinate of P_j. We would consider the vector from P_j to P_{j+1} to be a micro-stroke. On the other hand, it is also possible to describe T as a set of subsets:

$$T = \{S_i = \{(x_{ij}, y_{ij}, z_{ij}) | j = 1, 2, 3, \ldots, m\} | i = 1, 2, 3, \ldots, n\} \tag{2}$$

where S_i is a stroke. As a sub-sequence of T it also has temporal relations to other strokes; it follows stroke S_{i-1} and precedes stroke S_{i+1}. To identify strokes, we require a classification function F_c (see Equation 3) for points.

$$F_c(X_{P_b}) = \left\{ \begin{array}{lll} 0: & P_j \in \mathrm{S}_i, & P_{j+1} \in \mathrm{S}_i \\ 1: & P_j \in \mathrm{S}_i, & P_{j+1} \in \mathrm{S}_{i+1} \end{array} \right\} \tag{3}$$

Given the unpredictability of the deterministic relationship between strokes, we used two statistical thresholds for F_c, $ST_1 = \mu + \sigma$ (weak filter) and $ST_2 = \mu + 2\sigma$ (strong filter), assuming a normal distribution [10]. The reason that the parameter for Equation 3 is not simply P_j is because an isolated point in 3D space doesn't contain enough information to determine to which stroke it belongs. So we need an information-rich attribute X_{P_j} for each P_j; one which has been heavily used in pattern recognition and artificial vision is high curvature [11,12]. We chose to use a mature approach for measuring curvature known as k-cosine [13] (still popular due to its simplicity [14]) and two other methods that are similar in conception. The basic idea of this method is to take any point P_j then pick only two points, P_{j-k} and P_{j+k} (k points away from P_j). These three points are

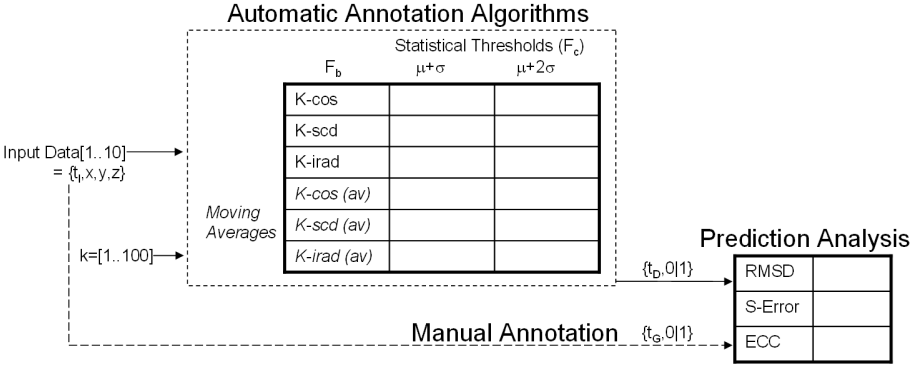


Fig. 1. Experimental Design

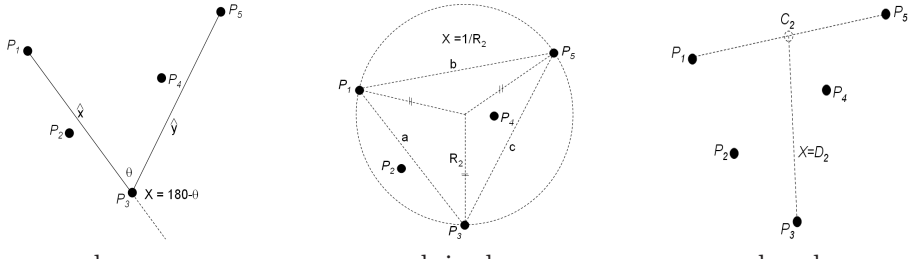
input to a boundary characterisation function F_b which produces the required information-rich attribute X_{P_j}:

$$X_{P_j} = F_b(P_{j-k}, P_j, P_{j+k}) \tag{4}$$

We use a factorial design (see Figure 1) for our experiment. We input 10 data sets (seven benchmark images and three realistic haptic data streams) which are then processed by F_b (our three k-metrics, with and without moving average [-k..k] ala [15]), then filtered using F_c (statistical thresholding). The predictive power of these automatic annotation algorithms is then analysed by several methods of mathematical comparison with a manually constructed ground truth, which we explore later in this paper.

3 K-Metrics

In this section we describe the three k-metrics for F_b we used in our investigation: cosine (k-cos), extrinsic curvature (k-irad), and symmetric centroid deviation (k-scd) (shown below for k=2)



k-cos k-irad k-scd

Since all of our methods were based on the general idea of k-cosine (k-cos), we describe it first. In this method, the boundary characterisation function F_b

is related to the cosine of the angle between the vectors (see Equation 5 [16]). The metric $X_{P_b} = 180 - \theta$. It accounts for point co-linearity: if $P_a = P_c$ then $X_{P_b} = 180$, otherwise if P_b lies directly between P_a and P_c then $X_{P_b} = 0$.

$$cos\theta = \frac{\hat{x} \cdot \hat{y}}{|x||y|} \tag{5}$$

Our second method, k-extrinsic curvature (k-irad), does more than consider the angle between three points - it also factors in the distance between the three points. The so-called *circumradius* (see Equation 6 [17]) can be trivially calculated between any three points that are non-colinear as shown below (also for k=2). The boundary characterisation function F_b is the extrinsic curvature (the inverse of the radius). This method accounts for point co-linearity in a slightly different way than k-cosine. If $P_a = P_c$ then the radius is half the distance from P_a to P_b, which is unbounded (as opposed to k-cosine). On the other hand, if P_b lies directly between P_a and P_c then the denominator of Equation 6 is zero so our algorithm uses an early check to set $X_{P_b} = 0$.

$$R = \frac{abc}{\sqrt{(a+b+c)(b+c-a)(c+a-b)(a+b-c)}} \tag{6}$$

Our third metric, k-symmetric centroid deviation (k-scd), simply calculates the centroid C_k between P_a and P_c then X_{P_b} is simply the distance between the centroid and P_b (see Equation 7). This method accounts for co-linearity similarly to k-extrinsic curvature. If $P_a = P_c$ then X_{P_b} is simply the distance from P_a to P_b. Else if P_b lies directly between P_a and P_c then $C_k = P_b$ so $X_{P_b} = 0$.

$$X = \sqrt{(c_k - p_i)^2} \tag{7}$$

In this section we described the three basic boundary characterisation functions F_b that will be investigated both with and without a moving average.

4 Assessing Predictive Performance

Which of our six methods makes better predictions than another? There are three popular mathematical methods for comparing predicted values with observed values: root mean squared deviation (RMSD); true-positive ratios; and mutual information measures. RMSD (see Equation 8) is a widely-used statistical measure of the difference between values predicted by a model D and the nearest values actually observed from the thing being modeled or estimated G [18].

$$RMSD(D,G) = \frac{\sqrt{\sum_{i=1}^{n}(x_{Di} - x_{Gi})^2}}{n} \tag{8}$$

However, perspective differences exist, depending on the parameters of this equation. For example, if the algorithm predicted every point as a stroke boundary, the RMSD from the perspective of the ground truth would be zero, but from the perspective of the prediction it would be very large. Thus we use:

$$\varepsilon = \max\{RMSD(D,G), RMSD(G,D)\} \tag{9}$$

RMSD is certainly a useful measure, a low RMSD means that predictions are accurate. However a high RMSD is ambiguous; it can mean that predictions generally are bad, or that predictions generally are reasonable but can be skewed by outliers. So, rather than just looking at averages, we considered other statistical measures relating accurate to inaccurate predictions (e.g. Equation 10 [19]).

$$\text{Sensitivity} = \frac{\text{No. of true positives}}{\text{No. of true positives} + \text{No. of false negatives}} \tag{10}$$

However we avoid such measures directly because they rely on choosing on a distance ϵ from the ground truth within which a prediction is considered true. Instead, with Equation 11 we calculate the maximum (ϵ) between all ground truth points and the corresponding closest data point which would give sensitivity = 1. Calculating the sensitivity error is the same as calculating the furthest outlier which would impact the RMSD, so a low sensitivity error means greater prediction stability.

$$\text{Sensitivity Error} = \max\{\forall G \forall D \min\{P_G(t) - P_D(t)\}\} \tag{11}$$

Finally, there are several information-theoretic methods able to compare how dependent G is on D considered as distributions. In the medical imaging literature a very popular method for comparing dependence is mutual information [20], we prefer a related measure called entropy correlation coefficient (ECC) [21] because ECC=0 means no dependence and ECC=1 means full dependence. In order to construct ECC we first find the entropy of D and G singly (see Equation 12), then find the joint entropy (see Equation 13).

$$H(D) = -\sum_{D} p_D \log_2 p_D \tag{12}$$

$$H(D,\, G) = -\sum_{D,\, G} p_{D,\, G} \log_2 p_{D,\, G} \tag{13}$$

Subsequently, we create a ratio measure relating the single and joint entropies called normalised mutual information (see Equation 14 [22]) then scale NMI into the range 0 to 1.

$$NMI(D,\ G) = \frac{H(D) + H(G)}{H(D,\ G)} \tag{14}$$

$$ECC(D,\ G) = 2 - \frac{2}{NMI(D,\ G)} \tag{15}$$

Thus we have three methods for comparing algorithms, RMSD, sensitivity error and entropy correlation co-efficient.

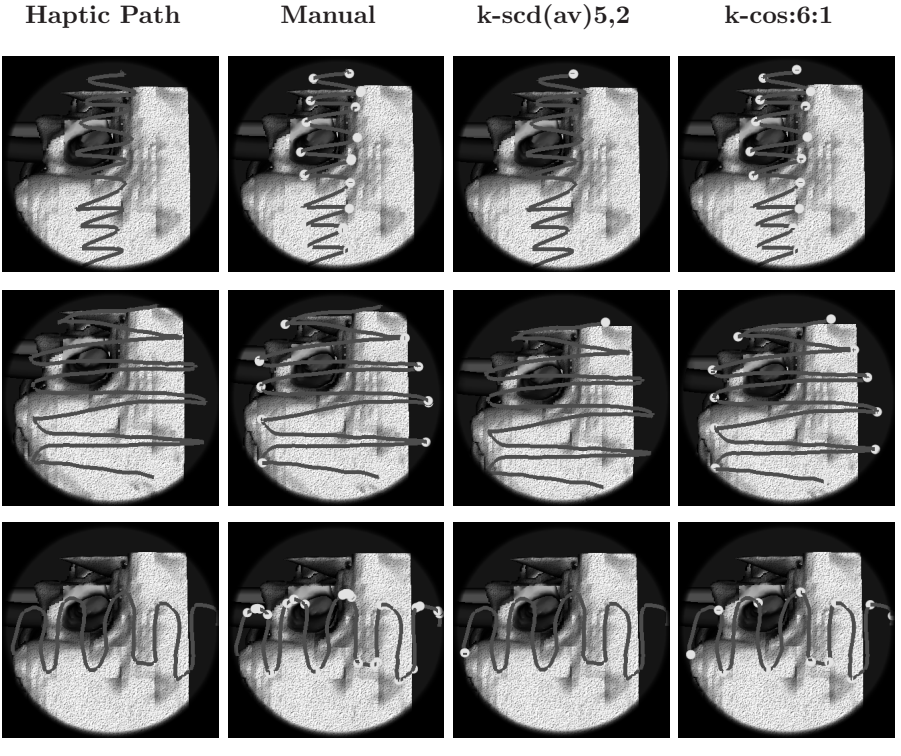
5 Results

We test our algorithms on two data sets (3D haptic and 2D benchmark) with our six k-metrics: K is distance from point of interest and ST is filter strength.

The *haptic* data set consists of three stroke types: fast (magnified), slow and wavy (non-overlapping thus amenable to screen-grab).

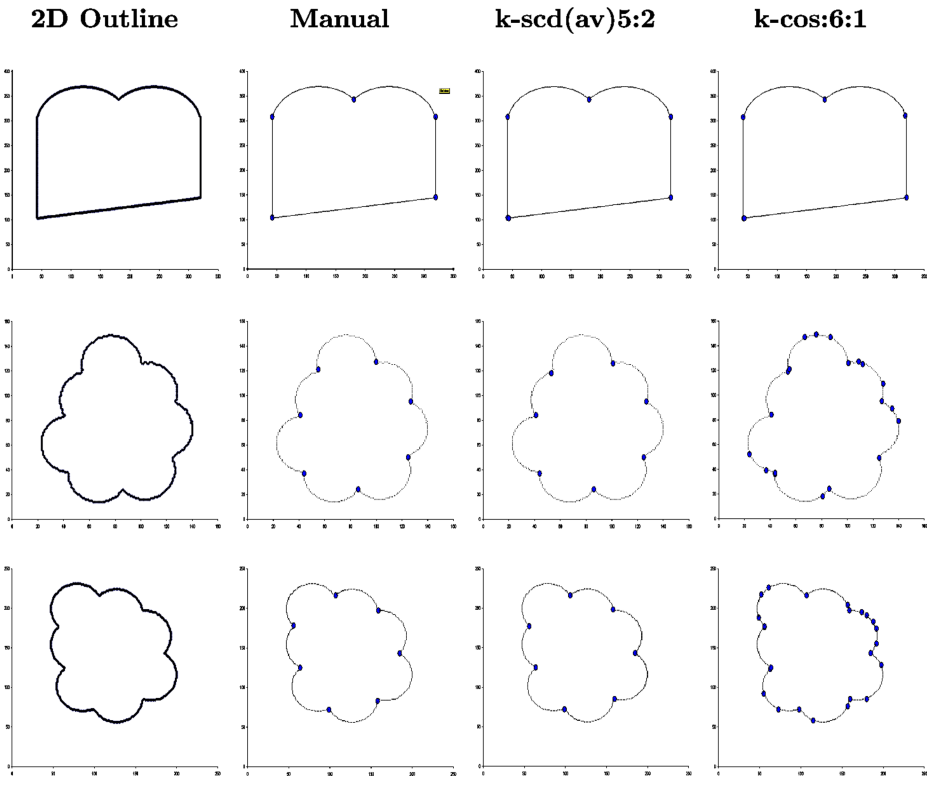
			RMSD		S-Error		ECC	
Haptic	K	ST	Mean	StD	Mean	StD	Mean	StD
k-cos	6	1	1.378	0.656	5.333	2.517	0.259	0.255
k-cos (av)	8	2	1.403	0.6	7	3.606	0.293	0.285
k-scd	4	2	6.367	0.777	85	19.16	0.01	0.004
k-scd (av)	5	2	6.817	1.268	94	26.85	0.009	0.005
k-irad	3	2	1.707	1.005	12.67	15.14	0.098	0.086
k-irad(av)	8	2	1.706	0.455	9.667	9.292	0.249	0.171

The four columns below show: hand motion, ground truth, then two example predictions made by the best performing methods for both data.

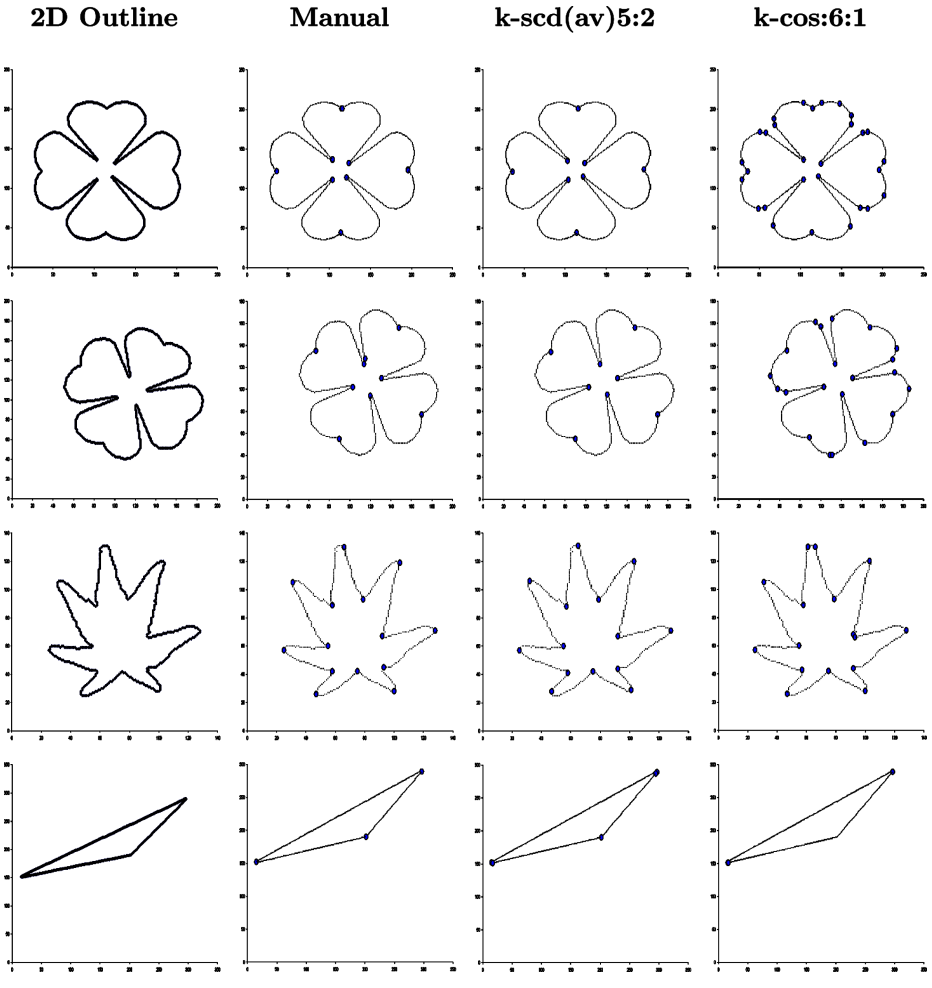


The *2D benchmark* data set consists of seven images used for testing corner detection methods [14]. These images were adapted to our temporal domain in the following way. First, the images were morphologically eroded by one pixel to shrink the original image I_1 by one pixel to make a new image I_2. Then I_2 was subtracted from I_1 to produce the image outline I_3. We then developed a simple edge crawling algorithm to convert I_3 into a time series of 2D points (which is exactly the same as T from Section 1 but all Z=0).

			RMSD		S-Error		ECC	
2D	K	ST	Mean	StD	Mean	StD	Mean	StD
k-cos	6	1	2.535	2.631	1.571	1.272	0.361	0.167
k-cos (av)	8	2	2.751	2.784	27.71	43.39	0.478	0.126
k-scd	4	2	2.448	2.757	2.571	1.902	0.408	0.217
k-scd (av)	5	2	2.454	2.737	2.429	2.07	0.475	0.232
k-irad	3	2	2.493	1.488	21.14	35.26	0.025	0.059
k-irad(av)	8	2	2.784	1.456	23.43	35.19	0.002	0.001



The algorithm that made the best predictions of the haptic data on average was k-cos, with a k value of 6 and a statistical threshold of $\mu+\sigma$, which produced a good average RMSD of 1.378 and a competitive ECC of 0.255. Visual comparison of the manually constructed haptic ground truth with k-cos:6:1 shows good face validity. With respect to the 2D benchmark images, k-cos:6:1 had comparatively poor face validity, often generating high numbers of false positives. K-scd:4:2 and k-scd(av):5:2 both had low RMSD on the 2D benchmark images (2.448 and 2.454 respectively), which we considered a tie. However k-scd(av):5:2 had a higher entropy correlation coefficient (0.475 as opposed to 0.408), so we considered it to be superior in the 2D case. Visual comparison of the manually



constructed ground truth for the 2D benchmark images shows good face validity for k-scd(av):5:2. Note that in this analysis we ranked RMSD above sensitivity error and ECC because of the greater meaningfulness of the magnitude of this single measure.

6 Conclusion

This paper presented and compared six k-metrics for classifying haptic data, in our case, motion recorded from a user of a virtual drill in a haptic-enabled temporal bone drilling simulation. Our preliminary results suggested that k-cos might be a competitive method for the automatic annotation of haptic motion in terms of strokes. Thus, this metric seems well suited for implementation in haptic simulations which require quantification of the relation between hand motion and performance. However, other experimental results showed that k-symmetric

centroid deviation was superior for the 2D benchmark images, thus perhaps for other two dimensional and relatively smooth data like sound waveform sampling and classification it would be preferable to use this method.

The ability to correctly identify strokes is beneficial for several reasons. Firstly, it allows hypotheses about differences between experts and novices with respect to stroke differences to be investigated within thousands of lines of haptic tool tracking data which would be infeasible to annotate manually. If such differences can be modelled, it should be possible to automatically assess surgical trainee performance, both for giving trainees feedback (realtime and otherwise) and also as part of a suite of trainee assessment tools.

Acknowledgments

We thank Matthew Hutchins and Chris Gunn from the CSIRO ICT Centre who initially worked on the temporal bone simulator with which we are continuing. We also thank Les Kitchen from Melbourne University whose comments and suggestions on a draft of this paper were invaluable.

References

1. Hutchins, M.A., O'Leary, S., Stevenson, D., Gunn, C., Krumpholz, A.: A networked haptic virtual environment for teaching temporal bone surgery. In: MMVR XIII, pp. 204–207. IOS Press, Amsterdam (2005)
2. Morris, D., Sewell, C., Barbagli, F., Salisbury, K., Blevins, N.H., Girod, S.: Visuohaptic simulation of bone surgery for training and evaluation. IEEE Comput. Graph. Appl. 26(6), 48–57 (2006)
3. Rasmussen, M., Mason, T.P., Millman, A., Evenhouse, R., Sandin, D.J.: The virtual temporal bone, a tele-immersive educational environment. Future Gen. Comp. Sys. 14(1-2), 125–130 (1998)
4. Wiet, G.J., Bryan, J., Dodson, E., Sessanna, D., Streadney, D., Schmalbrock, P., Welling, B.: Virtual temporal bone dissection simulation. In: MMVR 2000. IOS Press, Amsterdam (2000)
5. John, N.W., Thacker, N., Pokric, M., Jackson, A., Zanetti, G., Gobbetti, E., Giachetti, A., Stone, R., Campos, J., Emmen, A., Schwerdtner, A., Neri, E., Franceschini, S., Rubio, F.: An integrated simulator for surgery of the petrous bone. In: MMVR 2001. IOS Press, Amsterdam (2001)
6. Agus, M., Giachetti, A., Gobbetti, E., Zanetti, G., Zorcolo, A., John, N.W., Stone, R.J.: Mastoidectomy simulation with combined visual and haptic feedback. In: MMVR 2002. IOS Press, Amsterdam (2002)
7. Pflesser, B., Petersik, A., Tiede, U., Hohne, K., Leuwer, R.: Volume cutting for virtual petrous bone surgery. Computer Aided Surgery 7(2), 74–83 (2002)
8. Agus, M., Giachetti, A., Gobbetti, E., Zanetti, G., Zorcolo, A.: Tracking the movement of surgical tools in a virtual temporal bone dissection simulator. In: Ayache, N., Delingette, H. (eds.) IS4TM 2003. LNCS, vol. 2673. pp. 1004–1011. Springer, Heidelberg (2003)
9. Sewell, C., Morris, D., Blevins, N.H., Agrawal, S., Dutta, S., Barbagli, F., Salisbury, K.: Validating metrics for a mastoidectomy simulator. In: MMVR XV. IOS Press, Amsterdam (2007)

10. Kenney, J.F., Keeping, E.S.: Mathematics of Statistics Pt - 1, 3rd edn. Van Nostrand (1962)
11. Inesta, J.M., Buendia, M., Sarti, M.A.: Reliable polygonal approximations of imaged real objects through dominant point detection. Pattern Recogn. Lett. 31, 685–697 (1998)
12. Attneave, F.: Informational aspects of visual perception. Psychological Review 61, 183–193 (1954)
13. Rosenfeld, A., Johnson, E.: Angle detection on digital curves. IEEE Trans. Comput, 875–878 (1973)
14. Sun, T.H., Lo, C.C., Yu, P.S., Tien, F.C.: Boundary-based corner detection using k-cosine. Systems, Man and Cybernetics. In: ISIC. IEEE International Conference on (October 7-10), pp. 1106–1111 (2007)
15. Rosenfeld, A., Weszka, J.S.: An improved method of angle detection on digital curves. IEEE Trans. Comput. 24(9), 940–941 (1975)
16. Arfken, G.: Scalar or Dot Product, 3rd edn. Academic Press, London (1985)
17. Johnson, R.A.: Modern Geometry: An Elementary Treatise on the Geometry of the Triangle and the Circle. Houghton Mifflin (1929)
18. Maiorov, V.N., Crippen, G.M.: Significance of root-mean-square deviation in comparing three-dimensional structures of globular proteins. J Mol Biol 235(2), 625–634 (1994)
19. Kanji, G.K.: 100 Statistical Tests. SAGE Publications, Thousand Oaks (1999)
20. Pluim, J.P., Maintz, J.B., Viergever, M.A.: Mutual-information-based registration of medical images: a survey. IEEE Transactions on Medical Imaging 22, 986–1004 (2003)
21. Collignon, A., Maes, F., Vandermeulen, D., Marchal, G., Suetens, P.: Multimodality image registration by maximization of mutual information. Medical Imaging, IEEE Transactions on 16(2), 187–198 (1997)
22. Studholme, C., Hill, D., Hawkes, D.: An overlap invariant entropy measure of 3d medical image alignment. Pattern Recognition 1(32), 71–86 (1999)