

Features Extraction and Matching of Teeth Image Based on the SIFT Algorithm

Wang Xinzui

Suzhou Institute of Biomedical Engineering and
Technology

Dong Ningning

Suzhou Institute of Biomedical Engineering and
Technology

Li Huanli

Changchun Institute of Optics, Fine Mechanics and Physics,
Chinese Academy of Sciences
Graduate School of the Chinese Academy of Sciences

Abstract-Using of SIFT algorithm in the image of teeth model, can detect the features of the teeth image effectively. In this approach, first, search over all scales and image locations by using a difference-of-Gaussian function to identify potential interest points that are invariant to scale and orientation. Second, select keypoints based on measures of their stability and a detailed model is fit to determine location and scale at each candidate location. Third, assign one or more orientations to each keypoint location based on local image gradient directions. Last, measure the local image gradients at the selected scale in the region around each keypoint. And then use the KNN algorithm to match the features. Through lots of experiments and comparing with other feature extraction methods, this method can detect the features of the teeth model effectively, and offer some available parameters for 3D reconstruction of the teeth model.

Keywords-SIFT, feature extraction, teeth image

I. INTRODUCTION

In oral medicine, it is a very attractive technology to simulate the results before the treatment, which acquire the parameters of the treatment from the stimulation of treatment automatically. If we use the computer technology in oral medicine and reconstruct the three-dimensional model of the teeth through three-dimensional computer technology, the patients can see the detail of their oral cavity and teeth from different angles. It is very helpful for the doctors' treatment that simulating the correction process.

The indispensable part of the 3D reconstruction is that features extraction and matching. Features contain some important information of the image. For image matching and recognition, SIFT (Scale Invariant Feature Transform) features are first extracted from a set of reference images and stored in a database. A new image is matched by individually comparing each feature from the new image to this previous database and finding candidate matching features based on euclidean distance of their feature vectors. This paper will discuss fast nearest-neighbor algorithms that can perform this computation rapidly against large databases.

II. THE SIFT FEATURES EXTRACTION

A. Detection of scale-space extrema

The scale space of an image is defined as a function, $L(x, y, \sigma)$, that is produced from the convolution of a variable-scale Gaussian, $G(x, y, \sigma)$, an input image, $I(x, y)$:

$$L(x, y, \sigma) = G(x, y, \sigma) * I(x, y) \quad (1)$$

where $*$ is the convolution operation in x and y , and

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2)/2\sigma^2} \quad (2)$$

To efficiently detect stable keypoint locations in scale space, proposed using scale-space extrema in the difference-of-Gaussian function convolved with the image, $D(x, y, \sigma)$, which can be computed from the difference of two nearby scales separated by a constant multiplicative factor k :

$$\begin{aligned} D(x, y, \sigma) &= (G(x, y, k\sigma) - G(x, y, \sigma)) * I(x, y) \\ &= L(x, y, k\sigma) - L(x, y, \sigma) \end{aligned} \quad (3)$$

An efficient approach to the construction of $D(x, y, \sigma)$ is shown in Figure 1.

For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space images shown on the left of Figure 1. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images on the right of Figure 1. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated. The accuracy of sampling relative to σ is no different than for the start of the previous octave, while computation is greatly reduced.

In order to detect the local maxima and minima of $D(x, y, \sigma)$, each sample point is compared to its eight neighbors in the current image and nine neighbors in the scale above and below (see Figure 2). It is selected only if it is larger than all of these neighbors or smaller than all of them. The cost of this check is reasonably low due to the fact that most sample points will be eliminated following the first few checks.

B. Accurate keypoint localization

Once a keypoint candidate has been found by comparing a pixel to its neighbors, the next step is to perform a detailed fit to the nearby data for location, scale, and ratio of principal curvatures. This information allows

points that have low contrast (and are therefore sensitive to noise) or are poorly localized along an edge to be rejected.

Using the Taylor expansion (up to the quadratic terms) of the scale-space function, $D(x, y, \sigma)$, shifted so that the origin is at the sample point:

$$D(x) = D + \frac{\partial D^T}{\partial X} X + \frac{1}{2} X^T \frac{\partial^2 D}{\partial X^2} X \quad (4)$$

where D and its derivatives are evaluated at the sample point and $X = (x, y, \sigma)^T$ is the offset from this point. The location of the extremum, \hat{X} , is determined by taking the derivative of this function with respect to X and setting it to zero, giving

$$\hat{X} = -\frac{\partial^2 D^{-1}}{\partial X^2} \frac{\partial D}{\partial X} \quad (5)$$

The difference-of-Gaussian function will have a strong response along edges, even if the location along the edge is poorly determined and therefore unstable to small amounts of noise.

A poorly defined peak in the difference-of-Gaussian function will have a large principal curvature across the edge but a small one in the perpendicular direction. The principal curvatures can be computed from a 2×2 Hessian matrix, H , computed at the location and scale of the keypoint:

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix} \quad (6)$$

The derivatives are estimated by taking differences of neighboring sample points.

Let α be the eigenvalue with the largest magnitude and β be the smaller one. Then, we can compute the sum of the eigenvalues from the trace of H and their product from the determinant:

$$Tr(H) = D_{xx} + D_{yy} = \alpha + \beta \quad (7)$$

$$Det(H) = D_{xx}D_{yy} - (D_{xy})^2 = \alpha\beta \quad (8)$$

In the unlikely event that the determinant is negative, the curvatures have different signs so the point is discarded as not being an extremum. Let r be the ratio between the largest magnitude eigenvalue and the smaller one, so that $\alpha = r\beta$. Then,

$$\frac{Tr(H)^2}{Det(H)} = \frac{(\alpha + \beta)^2}{\alpha\beta} = \frac{(r\beta + \beta)^2}{r\beta^2} = \frac{(r+1)^2}{r} \quad (9)$$

which depends only on the ratio of the eigenvalues rather than their individual values. The quantity $(r+1)^2/r$ is at a minimum when the two eigenvalues are equal and it increases with r . Therefore, to check that the ratio of principal curvatures is below some threshold, r , we only need to check

$$\frac{Tr(H)^2}{Det(H)} < \frac{(r+1)^2}{r} \quad (10)$$

C. Orientation assignment

By assigning a consistent orientation to each keypoint based on local image properties, the keypoint descriptor can be represented relative to this orientation and therefore achieve invariance to image rotation.

The scale of the keypoint is used to select the Gaussian smoothed image, L , with the closest scale, so that all computations are performed in a scale-invariant manner. For each image sample, $L(x, y)$, at this scale, the gradient magnitude, $m(x, y)$, and orientation, $\theta(x, y)$, is precomputed using pixel differences:

$$m(x, y) = \sqrt{(L(x+1, y) - L(x-1, y))^2 + (L(x, y+1) - L(x, y-1))^2} \quad (11)$$

$$\theta(x, y) = \tan^{-1}((L(x, y+1) - L(x, y-1))/(L(x+1, y) - L(x-1, y))) \quad (12)$$

D. The local image descriptor

A keypoint descriptor is created by first computing the gradient magnitude and orientation at each image sample point in a region around the keypoint location. Figure 3 illustrates the computation of the keypoint descriptor. First the image gradient magnitudes and orientations are sampled around the keypoint location, using the scale of the keypoint to select the level of Gaussian blur for the image. In order to achieve orientation invariance, the coordinates of the descriptor and the gradient orientations are rotated relative to the keypoint orientation.

III. FEATURES MATCHING

The first step in the panoramic recognition algorithm is to extract and match SIFT features between all of the images. SIFT features are located at scale-space maxima/minima of a difference of Gaussian function. At each feature location, a characteristic scale and orientation is established. This gives a similarity-invariant frame in which to make measurements. Although simply sampling intensity values in this frame would be similarity invariant, the invariant descriptor is actually computed by accumulating local gradients in orientation histograms. This allows edges to shift slightly without altering the descriptor vector, giving some robustness to affine change. The vector of gradients is normalised, and since it consists of differences of intensity values, it is invariant to affine changes in intensity.

Assuming that the camera rotates about its optical centre, the group of transformations the images may undergo is a special group of homographies. We parameterise each camera by 3 rotation angles $\theta = [\theta_1, \theta_2, \theta_3]$ and focal length f . This gives pairwise homographies $\tilde{u}_i = H_{ij}\tilde{u}_j$

$$H_{ij} = K_i R_i R_j^T K_j^{-1} \quad (13)$$

Where

$$K_i = \begin{bmatrix} f_i & 0 & 0 \\ 0 & f_i & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (14)$$

and (using the exponential representation for rotations)

$$R_i = e^{[\theta_i]_{\times}}, \quad [\theta_i]_{\times} = \begin{bmatrix} 0 & -\theta_{i3} & \theta_{i2} \\ \theta_{i3} & 0 & -\theta_{i1} \\ -\theta_{i2} & \theta_{i1} & 0 \end{bmatrix} \quad (15)$$

However, for small changes in image position

$$u_i = u_{i0} + \left. \frac{\partial u_i}{\partial u_j} \right|_{u_{i0}} \Delta u_j \quad (16)$$

or equivalently $\tilde{u}_i = A_{ij} \tilde{u}_j$, where

$$A_{ij} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & 0 & 1 \end{bmatrix} \quad (17)$$

is an affine transformation obtained by linearising the homography about u_{i0} . This implies that each small image patch undergoes an affine transformation, and justifies the use of SIFT features which are partially invariant under affine change.

Once features have been extracted from all n images (linear time), they must be matched. Since multiple images may overlap a single ray, each feature is matched to its k nearest neighbours (we use $k = 4$). This can be done in $O(n \log n)$ time by using a k-d tree to find approximate nearest neighbours.

IV. THE RESULT OF THE SIFT FEATURES EXTRACTION AND MATCHING

Figure 4 shows the image of teeth models from three angles.

Figure 5 shows the SIFT features of each images. There have 756 features in (a), 1034 features in (b) and 700 features in(c).

Figure 6 shows the features match between each images.

There have 115 matches between(a)and(b), 33 matches between (a) and (c), 161 matches between (b) and (c).

It can be seen from the result: the SIFT method can extract many features, and the matching method can match

the features very well, there has little wrong matches, which can provide available parameters for the 3D reconstruction of the teeth.

V. CONCLUSIONS

This paper reported the using of the computer technology in oral medicine. Using the SIFT algorithm in the image of teeth model, which can detect the features in the teeth image effectively. Using the KNN algorithm to match the fetures, which can offer correct parameters for the three-dimensional reconstruction.

Suzhou science and technology plan project (YJS0952)
CAS knowledge innovation project (KGCX2-YW-911-2, KGCX2-YW-908-2, KGCX2-YW-911-1)

Corresponding author Xin-zui Wang, E-mail: xinzui_wang@163.com

REFERENCES

- [1] Yu-Jin Zhang, Image Engineering, 2007, Second Edition, Cambridge University Press, Beijing.
- [2] David G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, International Journal of Computer Vision, 2004.
- [3] Se, S., Lowe, D.G., and Little, J. 2002. Global localization using distinctive visual features. InInternational Conference on Intelligent Robots and Systems, IROS 2002, Lausanne, Switzerland.pp. 226-231.
- [4] Matthew Brown ,David G. Lowe, Automatic Panoramic Image Stitching using Invariant Features.

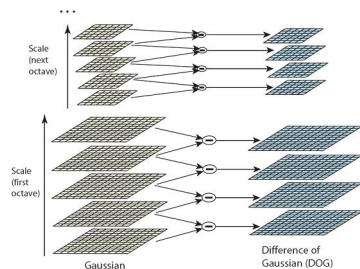


Figure1. the transformation from image space to scale space

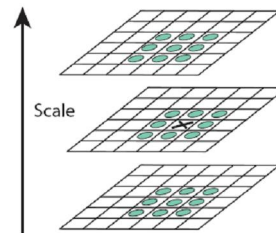


Figure 2. the extremum detection of the DoG scale space

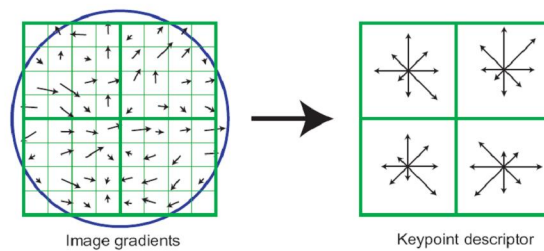


Figure 3. the computation of the keypoint descriptor

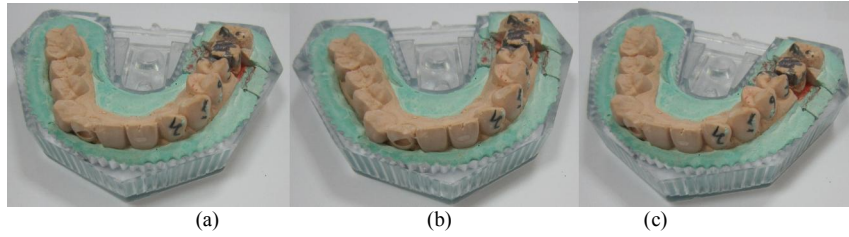


Figure 4. the image of teeth models form three angles

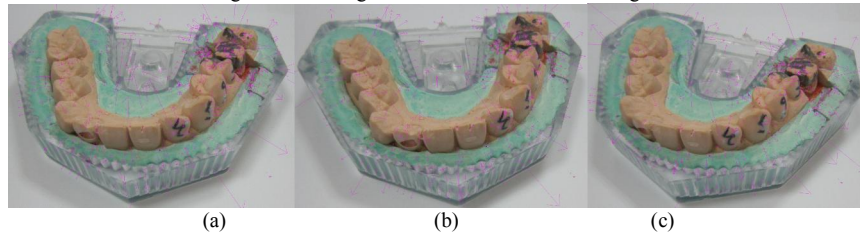


Figure 5. the SIFT features of each images

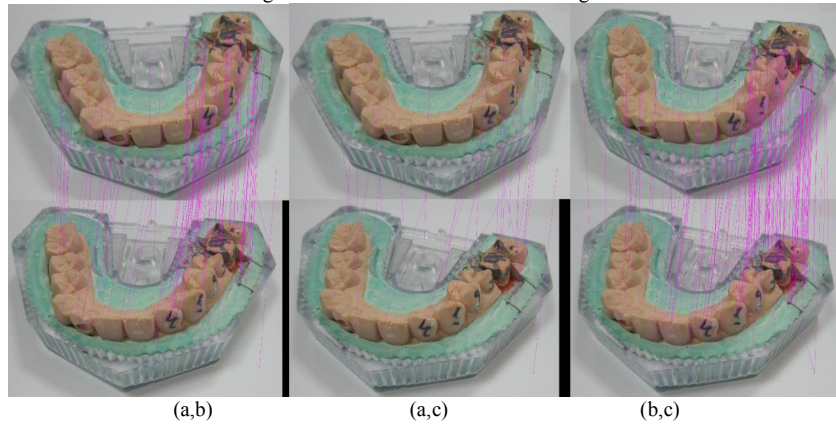


Figure 6. shows the features match between each images