Neural Process Lett (2018) 48:1261–1279
https://doi.org/10.1007/s11063-018-9816-6

AIFD Based 2D Image Registration to Multi-View Stereo Mapped 3D Models

Biao Zhao[1]

Published online: 14 March 2018

Abstract Multi-view stereo (MVS) map based 3D range reconstruction is to generate 3D ranges by analyzing the surrounding snapshots from different perspectives. Different to the traditional method which employing the expensive and difficult maintaining laser range devices to calibrate the range of the real 3D objects, MVS has achieved its success by seeking the geometrical correlations between the correspondences from the snapshot of different perspectives. The concerning of MVS keeps rising thanks to the fast development of digital maps and 3D printing. Several algorithms with regard to MVS has been well developed and achieved their success with regard to reconstruction of 3D ranges. Meanwhile, most of the algorithms were mainly focusing on the fusion and merging of different scenes and surface refinement. Less capability of the feature matching algorithms on the affine invariant images renders the current MVS algorithms need huge amount of images with tiny perspective differences. In this paper, we will propose a new MVS algorithm, deploying our previous published Affine Invariant Feature Descriptor (AIFD) to detect and match the correspondences from different perspectives and applying Homograph matrix and segmentation to define the planes of the objects. Thanks to the AIFD and Homograph based projection model, our proposed MVS algorithm outperform other MVS algorithms in terms of speed and efficiency.

Keywords AIFD · Feature matching · Homograph · Registration · Camera model · Multi-view Stereo

1 Introduction

Reconstructing 3D ranges from some snapshots of different perspective is a classical computer vision problem with ever existed concerning. Its applications range from 3D mapping, navigation to 3D printing, computational photography, video games, or heritage archival.

✉ Biao Zhao
biao_zhao@yahoo.com

[1] University of Lincoln, Lincoln, Brayford Way, Brayford Pool, Lincoln LN6 7TS, UK

Only recently have these techniques matured enough to provide industrial scale robustness, accuracy and scalability. The target of an image-based 3D reconstruction algorithm can be described as estimating the most likely 3D models by the given set of images under a proper assumption of material, viewpoints and light conditions. A general MVS pipeline includes:

- Image collection.
- Calibration for the difference of the camera setting of each image.
- Correspondences detection among images.
- Reconstruction the 3D ranges according to the geometrical correspondences.
- Optionally reconstruct the materials of the scene.

Most MVS algorithms focuse on the 2D scenes fusion, merging and refinement, to achieve a dense and accurate 3D ranges estimation. Meanwhile the DoG based feature matching algorithms, which are less capable of detecting and matching the features under an affine invariant environment, are making the 3D ranges reconstruction less efficient until granted with a great amount of snapshots from all the possible perspectives.

An accurate and dense correspondence matching plays an important role in the MVS: an accurate correspondence matching makes the protagonist in the camera matrix calibration, coherent the images patch of different coordinates; a dense correspondence matching can establish a density depth clouds, rendering it more accurate and easier to smooth and merge the 3D ranges surface. The matches to construct the 3D ranges are determined by the images registration. 3D ranges to 2D images registration is largely depends on an accurate 2D camera calibration with respect to the acquired geometric 3D model. An accurate camera position estimation can largely determine the quality of 3D range's construction. Thus the registration issue can be simplified as how to conduct the camera matrix in the framework of the projection model [13]. A large part of the recent success of MVS is due to the success of the underlying Structure from Motion algorithms that compute the camera parameters.

Camera calibration is the fundamental of MVS registration. It refers to a set of values describing a camera configuration, that is, camera pose information consisting of location and orientation, and camera intrinsic properties such as focal length and pixel sensor size. There are many different ways or models to parametrize this camera configuration. There exist many cues that can be used to calculate the camera parameters from images including: stereo correspondence, pre-settled devices, snapshot calibration etc. The method the algorithms applied can largely restrict the range of its application: a controlled MVS capture use diffuse lights and a turn table to collect the images, outdoor capture can capture series of images around a small-scale scenes, and crowd-sourcing from online photo-sharing websites. General speaking, the algorithm which is capable of tacking arbitrarily snapshots are more desirable. In this situation, a robust, dense and accurate correspondence detecting and matching schema becomes quite necessary for the MVS.

Different to the traditional DoG based feature marching algorithm, we propose a novel MVS method utilizing our previous published AIFD to detect and match the correspondence from images to images and introducing the Homograph model to define the smooth planes from the 3D objects. AIFD is a feature detector and descriptor method, provided a more improved resilience to affine and scale invariance. It borrowed some ideas of SIFT [14], like scale space and pyramid structure, etc, but it is more capable to dealing with the image content of different view points, which is suits the special requirement of MVS.

Scale invariant feature detector, like SIFT, SURF [6], ALP [4], etc, has achieved its success on a lot of applications, including content-based visual retrieval, robotic navigation, image registration, etc. However, its sensitivity to the view point changes greatly restrict its applications to a larger range, such as 3D registration for instance for a long time. Borrowed

the basic principles of SIFT, we have successfully proposed Affine Invariant Feature Detector (AIFD), which has a better resilience to affine transformations. Equipped with this more advanced affine invariant feature detector, we can now seek the connections between images for 3D ranges reconstruction by detecting the matched features.

A correct detected correspondences between two images constitute a stereo system, which can provide the depth information. A set of points defined by the depth information outshapes the structure of the 3D ranges. In practice, most scenes or partials of the object will be covered by more than 2 images, which can help to calibrate a more dense and accurate spatial information. The origins of multi-view stereo can be traced back to human stereopsis and the first attempts to solve the stereoscopic matching problem as a computation problem [4]. Until today, two-view stereo algorithms have been a very active and fruitful research area . The multi-view version of stereo originated as a natural improvement to the two-view case. Instead of capturing two images from different perspectives, multi-view stereo would capture more viewpoints in-between to increase robustness, e.g. to image noise or surface texture and viewpoint. What started as a way to improve two-view stereo has nowadays evolved into a different type of problem.

Only equipped with sufficient amount of correspondences from different images, are we able to be more approach to an accurate camera matrix estimation by DLT. The knowledge of the registration from 3D ranges to 2D images can improve to map the 3D textures and in advance can be treated as a homograph reference to be apply to some other 2D images [3]. With this iterative registration-mapping method, the 3D ranges can be more precisely registered to an un-calibrated and arbitrarily snapshot [1].

Based on our proposed pipeline, a progressive mapping and registration 3D to 2D images registration method can be formed. The experiments in the below section can prove the performance of our proposed registration method outperforms against the traditional edge/corner based method [2]. Though our registration proposal, the stereo mapped 3D model can be introduced to more applications for its efficiency and simplicity [5].

2 Stereo Visual Based 3D Range Methods

3D range model is referring to a collection of points that presenting the distance in a scene from a specified viewpoint, which is normally associated with some type of sensor, like the Laser deceives [7]. To a well formed range model, its pixel value reflects the corresponding distance to a certain view plain[8]. If the sensor that is used to produce the range is properly calibrated, the pixel values can directly give the distance in physical units, like meter [9].

The sensor device that is used for producing the range model is sometimes referred to as a range camera. Range cameras can operate according to a number of different techniques [10], including Stereo triangulation, Sheet of light triangulation, Time-of-flight, Structured light, Interferometry, and Coded aperture. Sheet of light triangulation is achieved by changing of the scene illuminated with a sheet of light this creates a reflected line as seen from the light source. From any point out of the plane of the sheet the line will typically appear as a curve, the exact shape of which depends both on the distance between the observer and the light source, and the distance between the light source and the reflected points. By observing the reflected sheet of light using a camera (often a high resolution camera) and knowing the positions and orientations of both camera and light source, it is possible to determine the distances between the reflected points and the light source or camera. By illuminating the scene with a specially designed light pattern, structured light [11], depth can be determined using only a single image of the reflected light. The structured light can be in the form

of horizontal and vertical lines, points or checker board patterns. The depth can also be measured using the standard time-of-flight (ToF) technique, more or less like a radar, in that a range image similar to a radar image is produced, except that a light pulse is used instead of an RF pulse. By illuminating points with coherent light and measuring the phase shift of the reflected light relative to the light source it is possible to determine depth. Under the assumption that the true range image is a more or less continuous function of the image coordinates, the correct depth can be obtained using a technique called phase-unwrapping. Depth information may be partially or wholly inferred alongside intensity through reverse convolution of an image captured with a specially designed coded aperture pattern with a specific complex arrangement of holes through which the incoming light is either allowed through or blocked.

Among all of these techniques, Stereo triangulation is the most popular and widely applied technique for 3D ranges detections, where the depth data are determined by the data acquired by stereo or multiple-camera system. This way it is can determine the depth of a certain points in the scene, for example, from the enter point of the line between their focal points. In order to solve the depth measurement by employing a stereo camera system, it is necessary to detect the corresponding points from the different images. A well solution correctly specifying the correspondences from different images is one of the main task by applying this type of technique. For instance, it is difficult to detect the correspondence for the image points that lie inside the regions of homogeneous intensity or color. As a consequence, 3D range based stereo triangulation can produce reliable depth estimation only for a subset of all points visible from a multiple-view cameras.

The advantage of this technique is that it can guarantee that the measurement is passive and does not require a special requirement for detecting scene illumination. Other mentioned technique do not have to tangle the correspondence but are depending on some particular scene illuminations.

To a stereo image, a pixel value is not to store the grey scale level at a certain position, but to identify the depth information. Stereo cameras captures two images with tiny difference on the scene. In Fig. 1 the point A on the right light from is transmitted through the entry points of a pinhole cameras at B and D, onto image screens at E and H. In the attached diagram the distance between the centers of the two camera lens is $BD = BC + CD$. The triangles are similar, ACB and BFE and ACD and DGH.

$$\begin{aligned} \text{Therefore displacement } d &= EF + GH \\ &= \frac{BC + CD}{AC} \\ &= \frac{BD}{AC} \\ &= \frac{k}{z}, \text{ where } k = BD, z = AC \end{aligned} \tag{1}$$

Assuming the cameras are from the same level, and image plane is flat on the same plane, the displacement in the Y axis between the same pixel in the two images is $d = \frac{k}{z}$, Where k is the distance between the two cameras times the distance from the lens to the image. The depth component in the two images are z_1, z_1 given by,

$$\begin{aligned} z_2(x, y) &= \min \left\{ v : v = z_1 \left(x, y - \frac{k}{z_1(x, y)} \right) \right\} \\ z_1(x, y) &= \min \left\{ v : v = z_2 \left(x, y + \frac{k}{z_2(x, y)} \right) \right\} \end{aligned} \tag{2}$$

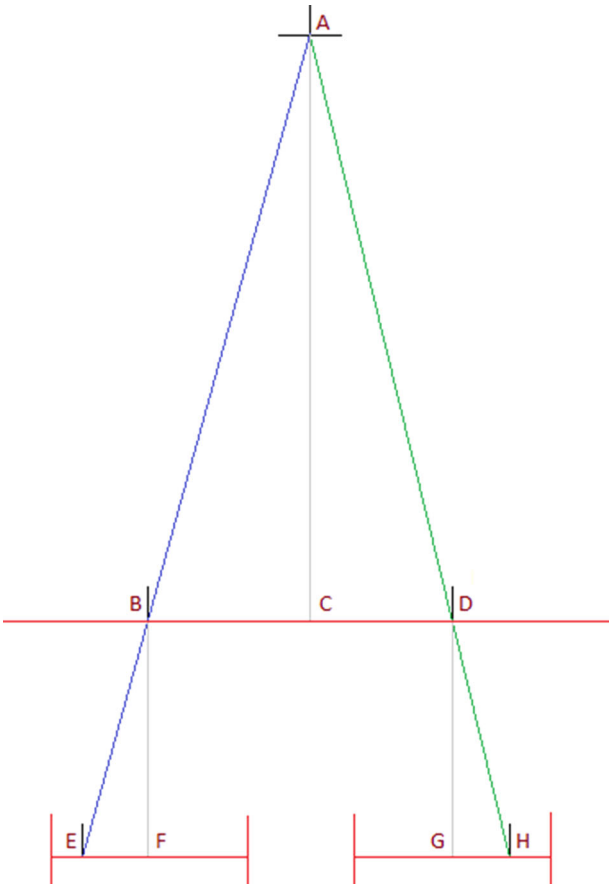

Fig. 1 Relationship of image displacement to depth with stereoscopic images, assuming flat co-planar images

Because the computation is squaring of the points in SSD, many implementations use Sum of Absolute Difference (SAD) as the basis for computing the measurement. Other methods use normalized cross correlation (NCC). The least squares measure can be deployed to measure the information content of the stereoscopic images, given the depth information at each point $z(x, y)$. The information needed to represent the image is called I_m. Image rectification is required to adjust the images as if they were co-planar. This may be achieved by a linear transformation. The images also need to be rectified to make each image equivalent to the one taken from a pinhole camera as if they are projected to a flat plane.

The normal distribution is,

$$P(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \tag{3}$$

Probability is related to information content described by message length L,

$$\begin{aligned} P(x) &= 2^{-L(x)} \\ L(x) &= -\log_2 P(x) \end{aligned} \tag{4}$$

So,

$$L(x, \mu, \sigma) = \log_2(\sigma\sqrt{2\pi}) + \frac{(x-\mu)^2}{2\sigma^2}\log_2 e \tag{5}$$

The least squares measure may be used to measure the information content of the stereoscopic images [], given depths at each point $z(x, y)$. Firstly the information needed to express one image in terms of the other is derived. This is called I_m.

A color difference function can be used to fairly measure the difference between colors. The color difference function is written cd in the following. The measure of the information needed to record the color matching between the two images is,

$$I_m(z_1, z_2) = \frac{1}{\sigma_m^2}\sum_{x,y} \mathrm{cd}\left(\mathrm{color}_1\left(x, y + \frac{k}{z_1(x, y)}\right), \mathrm{color}_2(x, y)\right)^2 \tag{6}$$

An assumption is made about the smoothness of the image. Assuming that two pixels are more likely to be of the same color, the closer the voxels they represent are. This measure is intended to favor colors that are similar being grouped at the same depth. For example, if an object in front occludes an area of sky behind, the measure of smoothness favors the blue pixels all being grouped together at the same depth. T he total measure of smoothness uses the distance between voxels as an estimate of the expected standard deviation of the color difference,

$$I_s(z_1, z_2) = \frac{1}{2\sigma_h^2}\sum_{i:\{1,2\}}\sum_{x_1,y_1}\sum_{x_2,y_2} \frac{\mathrm{cd}(\mathrm{color}_i(x_1, y_1), \mathrm{color}_i(x_2, y_2))^2}{(x_1-x_2)^2 + (y_1-y_2)^2 + (z_i(x_1, y_1) - z_i(x_2, y_2))^2} \tag{7}$$

The total information content is then summed as,

$$I_t(z_1, z_2) = I_m(z_1, z_2) + I_s(z_1, z_2) \tag{8}$$

The z component of each pixel must be chosen to provide the minimum value for the information content. This will give the most likely depths at each pixel. The minimum total information measure is,

$$I_{\min} = \min\{i : i = I_t(z_1, z_2)\}\} \tag{9}$$

The depth functions for the left and right images are the pair,

$$(z_1, z_2) \in \{(z_1, z_2) : I_t(z_1, z_2) = I_{\min}\} \tag{10}$$

3 Orthogonal Projection

Orthographic projection (sometimes orthogonal projection), is a means of representing three-dimensional objects in two dimensions. It is a form of parallel projection, in which all the projection lines are orthogonal to the projection plane,[1] resulting in every plane of the scene appearing in affine transformation on the viewing surface. The obverse of an orthographic projection is an oblique projection, which is a parallel projection in which the projection lines are not orthogonal to the projection plane.

The term orthographic is sometimes reserved specifically for depictions of objects where the principal axes or planes of the object are also parallel with the projection plane, [1] but these are better known as multi-view projections. When the principal planes or axes of

an object are not parallel with the projection plane, but are rather tilted to reveal multiple sides of the object, the projection is called an axonometric projection. Sub-types of multi-view projection include plans, elevations and sections. Sub-types of axonometric projection include isometric, dimetric and trimetric projections.

A simple orthographic projection onto the plane z = 0 can be defined by the following matrix:

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{11}$$

For each point $v = (vx, vy, vz)$, the transformed point would be

$$P_v = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ 0 \end{bmatrix} \tag{12}$$

Often, it is more useful to use homogeneous coordinates. The transformation above can be represented for homogeneous coordinates as

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{13}$$

For each homogeneous vector $v = (vx, vy, vz, 1)$, the transformed vector would be

$$Pv = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} v_x \\ v_y \\ v_z \\ 1 \end{bmatrix} = \begin{bmatrix} v_x \\ v_y \\ 0 \\ 1 \end{bmatrix} \tag{14}$$

4 AIFD Based Feature Matching

Our previously published feature descriptor AIFD achieves its resilience to affine and scale changes by reshaping the multi-scale image representation and local extrema detection in order to maintain a linear relationship under the changes of affine transformation. Instead of relying on the image simulations, AIFD achieves its affine and scale invariance completely based on its internal mechanisms when dealing with the transformed visual content. Thus it is more brief, reliable and feasible to more applications and has more potentials for future research.

It is based on our previous proposed affine scale space. For a given image $I(x, y)$, its scale space is given by a family of pre-smoothed images $L(x, y, \sigma)$, where the scale parameter is pre-defined according to its kernel size:

$$\begin{aligned} g(x, y; \sigma) &= \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \\ \text{such as,} \quad L(x, y; \sigma) &= g(x, y; \sigma) * I(x, y) \end{aligned} \tag{15}$$

Based on this definition, a structure more adaptable to the affine transformation is defined below:

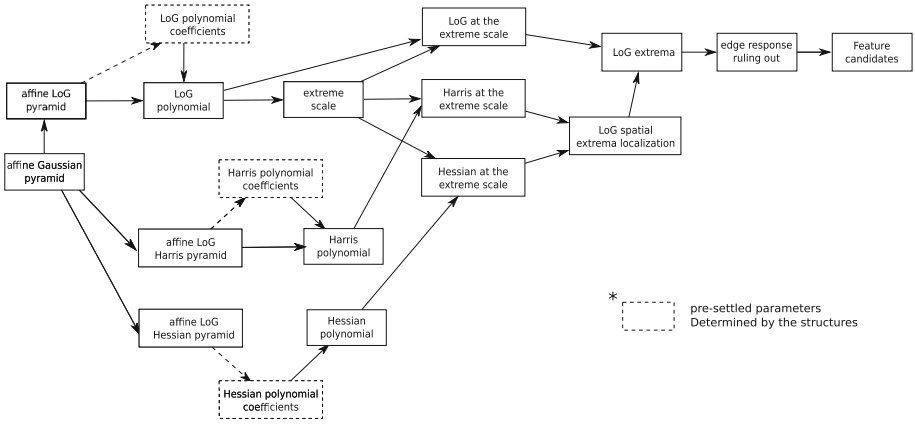

Fig. 2 The pipeline of AIFD. The parameters in the dashed box is pre-settled and only determined by the pyramid structure

$$g(\eta, \Sigma_s)_A = \frac{1}{2\pi\sqrt{det\,\Sigma_s}} e^{-\frac{\eta^T \Sigma_s^{-1} \eta}{2}}. \tag{16}$$

where, $\Sigma_s = A\sigma^2 A^T$.

In this formula, A represents the affine transformation, a 2×2 matrix. σ is the scale. This deformed Gaussian kernel is specialized to generate affine scale space which can maintain linear relationship regardless the change of view point. Based on this structure, the images from any view points can be well represented from multi-scales. From the definition of affine scale space, conventional isotropic scale space can be deemed as a spacial case, whose affine transformation equals to the 2×2 identity matrix (Fig. 2).

The similarity of two visual content is largely depends on the matched features detected from the scale space. To the conventional scale space, several approaches to detect the local maximum or minimum from derivatives have been proposed [12], and local LoG extrema detection outperform all others, concerning the accuracy and efficiency of a method in practice [17].

The Laplacian of Gaussian (LoG) scale space can be mathematically expressed as:

$$\nabla^2 L = L_{xx} + L_{yy} \tag{17}$$

In this formula, L represents the scale space. The local maximum or minimum over the Laplacian can then be selected as the feature candidates.

Borrowing the idea of LoG, we have also proposed an affine LoG, with the purpose to promote the feature candidates detection over affine scale space. Instead of a direct laplacian operation, we have proposed a feasible implementation based on our proposed pyramid structure to efficiently generate affine LoG. By this implementation, the affine Gaussian and LoG scale space can be simultaneously generated. More information about affine scale space and affine LoG can be found [18].

Once obtaining the affine scale space, we can then detect the local extrema by comparing its Harris and Hessian matrix. If we suppose,

$$A = \begin{bmatrix} \frac{\partial^2 f}{\partial x^2} & \frac{\partial^2 f}{\partial x \partial y} \\ \frac{\partial^2 f}{\partial x \partial y} & \frac{\partial^2 f}{\partial y^2} \end{bmatrix} \qquad B = \begin{bmatrix} \left(\frac{\partial f}{\partial x}\right)^2 & \frac{\partial f}{\partial x}\frac{\partial f}{\partial y} \\ \frac{\partial f}{\partial x}\frac{\partial f}{\partial y} & \left(\frac{\partial f}{\partial y}\right)^2 \end{bmatrix} \tag{18}$$

Nominating the eigenvalues of A as ψ_1 and ψ_2 and the eigenvalues of A as ψ_1 and ψ_2, we can then detect the features by analysing,

$$\frac{1}{4}\min\{\psi_1, \psi_2\}^2 > \max\{\nu_1, \nu_2\} \tag{19}$$

With this way to find the feature candidates, we can apply the LoG first and second derivative filters to form the Harris and Hessian matrices.

$$\begin{aligned} l_1 =& A^{-1}\eta \frac{1}{\pi\sigma^6} e^{-\frac{\eta^T (AA^T)^{-1}\eta}{2\sigma^2}} \left(2 - \frac{\eta^T (AA^T)^{-1}\eta}{2\sigma^2}\right) \\ l_{axx} =& \frac{1}{\pi\sigma^6}\left[\left(\frac{\eta^T (AA^T)\eta}{2\sigma^2} - 3\right)\left(\frac{M_a(1,1)}{\sigma^2} - 1\right) - 1\right] \\ l_{ayy} =& \frac{1}{\pi\sigma^6}\left[\left(\frac{\eta^T (AA^T)\eta}{2\sigma^2} - 3\right)\left(\frac{M_a(2,2)}{\sigma^2} - 1\right) - 1\right] \\ l_{axy} =& \frac{1}{\pi\sigma^6}\left(\frac{\eta^T (AA^T)\eta}{2\sigma^2} - 3\right)\frac{M_a(1,2)}{\sigma^2} \end{aligned} \tag{20}$$

Applying the same structure as depicted in Fig. 3 to generate the LoG derivative scale space to detect the local extrema.

Borrowing the idea of CDVS [16], we have proposed a polynomial expression to represent the affine scale space (also affine LoG and affine LoG derivatives) with a continual scale from. Supposing the the continual scale space can be approached by a cubic polynomial, which can be written as:

$$L(\sigma) = a \cdot \sigma^3 + b \cdot \sigma^2 + c \cdot \sigma + d \tag{21}$$

where, the parameters a, b, c, d are of the scale space's image size. σ represents a scale value within the Octave range. $L(\sigma)$ represents the image of a specified scale, it can be one image of Gaussian scale space, LoG scale space or LoG derivatives of any scale. This expression can be deployed as the continual form to represent the scale space of Gaussian, LoG or LoG derivatives. Apparently, σ can be any value within the Octave range, and $L(\sigma)$ represents the corresponding scale space. According to the expression above, the image of any specified scale can be calculated given the parameters a, b, c, d, which are related with the input image. Thanks to the pyramid structure, 4 scale space within each Octave can easily be generated.

The equations at the 4 parameters can be written as:

$$\begin{bmatrix} a(x, y) \\ b(x, y) \\ c(x, y) \\ d(x, y) \end{bmatrix} = M \begin{bmatrix} L(\sigma_1) \\ L(\sigma_2) \\ L(\sigma_3) \\ L(\sigma_4) \end{bmatrix}, \tag{22}$$

where M is a 4×4 matrices and a, b, c, d are linear combination of $L(\sigma_1)$, $L(\sigma_2)$, $L(\sigma_3)$ $L(\sigma_4)$. Matrix M is a simple 4×4 matrix and it is determined under the certain affine transformation. Thanks to the linear property of the polynomial expression, the matrix M

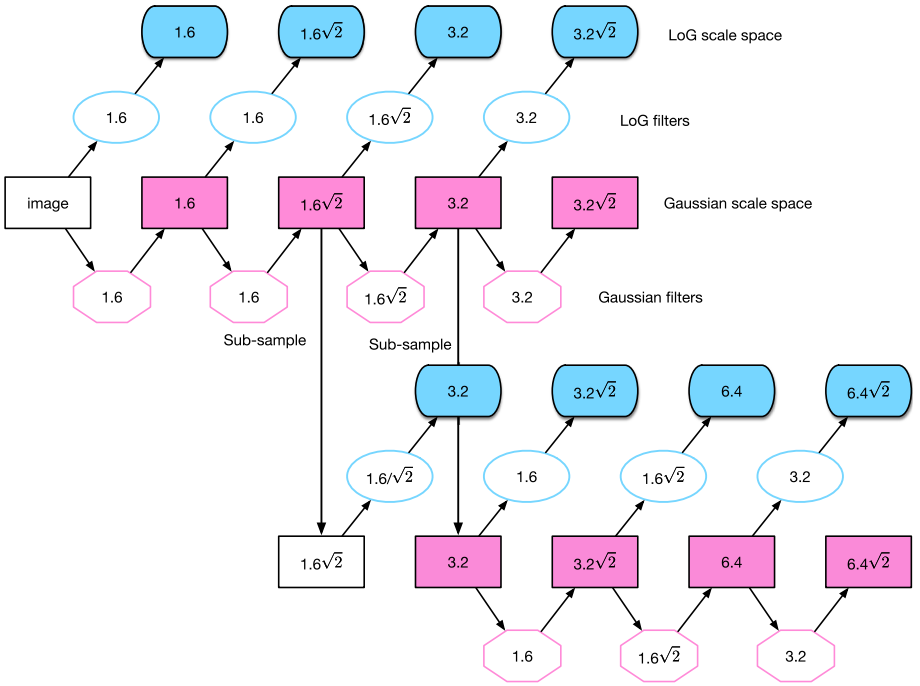

Fig. 3 Proposed pyramid structure to speed up the generation of Gaussian, LoG and also LoG derivatives scale space

will not be affected under a different view point, which can be demonstrated, if supposing the affine transformation is A.

$$L(A,\sigma) = \left[\sigma^3\ \sigma^2\ \sigma\ 1\right] M \begin{bmatrix} L(A,\sigma_1) \\ L(A,\sigma_2) \\ L(A,\sigma_3) \\ L(A,\sigma_4) \end{bmatrix} \tag{23}$$

The candidates, with the Harris-Hessian matrices subtraction smaller than 0.001 will also be rejected to guarantee the extrema is larger upto a certain level compared with the surrounding points by $R = Tr(H)^2/Det(H)$, which equals to $(\gamma+1)^2/\gamma$.

Since $L(\boldsymbol{x}+\boldsymbol{\Delta x})$ is the local extreme, its derivative equals $\mathbf{0}$. By taking the derivative on both sides of the equation, we can have:

$$\Delta \boldsymbol{x} = -\{D^2 L(\boldsymbol{x})\}^{-1} DL(\boldsymbol{x}) \tag{24}$$

If the offset is larger than 0.5 in any dimension, it implies the real extreme location is closer to a different pixel sample. Considering the local extreme scale for each pixel sample can be quite diverse, the feature located at the specified integral position becomes no more adequate.

The offset $\hat{x}$ will be summed up with the detected integral position to approach the local extreme location to sub-pixel's precision, according to the formula Eq. 31. The Hessian matrix and local gradient of the pixel sample can be obtained by the corresponding LoG derivative polynomial expressions.

To cope with an affine gradient-based feature descriptor, we will introduce the affine gradient filter and the related scale space in this section. At the very beginning, the definition of image gradient can be given by:

$$\nabla I = \left(\frac{\partial I}{\partial x}, \frac{\partial I}{\partial y} \right) \tag{25}$$

It is equivalent to the first order of image derivatives. The traditional method to calculate the image gradient is by taking the subtraction from two image neighbouring pixels in the form:

$$\nabla I = \frac{1}{2} \left(I(x-1, y) - I(x+1, y), I(x, y-1) - I(x, y+1) \right) \tag{26}$$

For every point of image gradient, its direction and magnitude can be given by:

$$\begin{aligned} m(x, y) &= \sqrt{\left(\frac{\partial I}{\partial x} \right)^2 + \left(\frac{\partial I}{\partial y} \right)^2}, \\ \theta &= \arctan \left(\frac{\frac{\partial I}{\partial x}}{\frac{\partial I}{\partial y}} \right). \end{aligned} \tag{27}$$

where I is the original image, $*$ is convolution operation, g is Gaussian filter. Thus, the derivative of the Gaussian blurred image of the corresponding scale space can be equivalently obtained though filtering the image with the Gaussian derivative filters, which can be derived by:

$$\begin{bmatrix} g_x \\ g_y \end{bmatrix} = \begin{bmatrix} \frac{\partial g}{\partial x} \\ \frac{\partial g}{\partial y} \end{bmatrix} = \begin{bmatrix} \frac{x}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \\ \frac{y}{2\pi\sigma^4} e^{-\frac{x^2+y^2}{2\sigma^2}} \end{bmatrix} = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} \begin{bmatrix} \frac{x}{\sigma^2} \\ \frac{y}{\sigma^2} \end{bmatrix} \tag{28}$$

Thus, the Gaussian derivatives scale space can easily be obtained by multiplying the corresponding Gaussian scale space with x/σ^2 and y/σ^2.

Borrowing the idea of affine LoG, we can have the affine Gaussian derivative filters as:

$$\begin{bmatrix} g_x^A \\ g_y^A \end{bmatrix} = g_\eta^A = \frac{A^{-1}\eta}{2\pi\sigma^4} e^{-\frac{\eta^T (AA^T)^{-1} \eta}{2\sigma^2}} \tag{29}$$

where A is a 2×2 matrix, indicating the affine transformation, $\eta = A \begin{bmatrix} x \\ y \end{bmatrix}$. Like the isotropic Gaussian derivative scale space, the affine Gaussian derivative scale space can also be generated by multiplying the corresponding Gaussian scale space with $A^{-1}\eta/\sigma^2$.

Gradients from affine transformed images are restrained by the affine matrices between different viewpoints. Around each feature, the relocated gradients, according to the affine transformation, can then form a histogram as the feature descriptor (Figs. 4, 5).

The gradient relocation can be done in the form of:

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = A \begin{bmatrix} x \\ y \end{bmatrix} \tag{30}$$

where x, y is the index of the gradient around the detected features from 1 to the descriptor, pre-defined by the scale of the features. x' and y' are the indexes accounting for the affine transformation. The relocation of the gradient can then be collected according to the new calculated index. The interpolation of the gradients may also be applied if the new calculated indexes are not integers.

Assigning an orientation to each feature, the feature descriptor can be represented relative to this orientation and achieve its invariance to image rotation. To calibrate the orientation

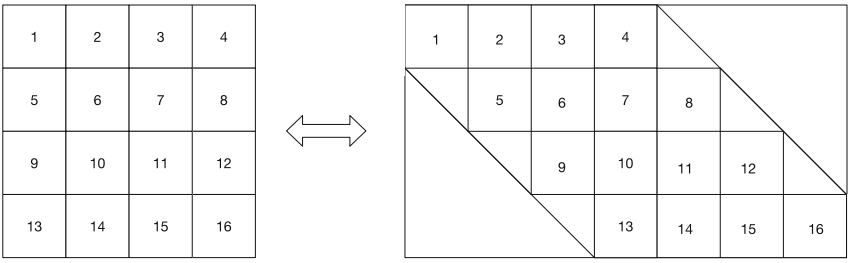

Fig. 4 Gradient relocated from the specified area around the detected features. The index of the gradient can be calculated according to the affine transformations

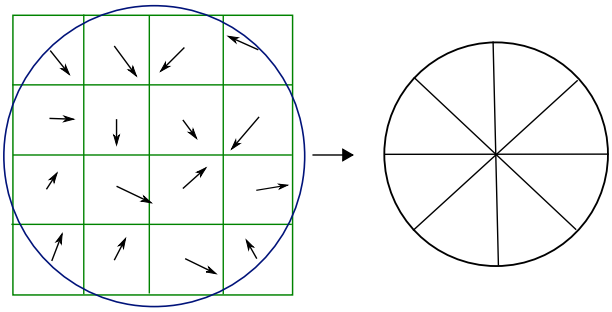

Fig. 5 The gradient around the feature will be added to the gradient histogram weighted by its magnitude and a Gaussian-weighted circular window [14]

of a feature, an area of scale space gradient around the feature will first be formed, after proceeding our proposed gradient relocation, eliminating the effect of the affine distortion. The area of gradient to be collected is in a square shape with its size equal to 3 times that of the feature scale. Then, the orientation of each sample of scale space gradient can be added to the orientation histogram weighted by its gradient magnitude and by a Gaussian-weighted circular window with 1.5 times the scale [14].

Then, the orientation histogram will be subdivided into 36 bins covering the 360° range of orientations and filled the corresponding accumulated magnitude. The peak of the histogram points to its main direction. Any other local peak that is within 80% of the highest peak and higher than the average of its two neighbors will be assigned with different orientations. The features with multiple peaks will be respectively created at the same location with the same scale but different orientations [14].

In advance, the main directions of the feature can also promote its accuracy by utilizing the second order of Taylor expansion in the form of:

$$\Delta \boldsymbol{x} = -\{D^2 L(\boldsymbol{x})\}^{-1} DL(\boldsymbol{x}) \tag{31}$$

Supposing the detected main direction bin is M and its two neighbor direction bins are M_- and M_+, the above equation can be implemented in the form of:

$$\Delta M = -\frac{\frac{1}{2}(M_+ - M_-)}{\frac{1}{2}(M_+ + M_-) - M} \tag{32}$$

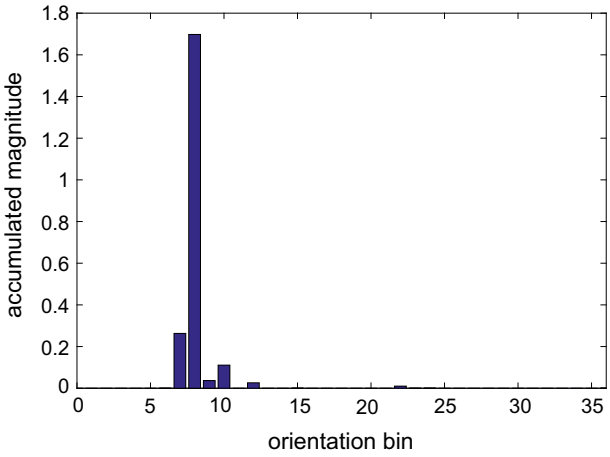

Fig. 6 Orientation histogram, created by subdividing the surrounding gradient into 36 bins according to the gradient orientations and accumulating weighted magnitude. The largest bin will be selected as the main direction of the descriptor

Then, the main direction is:

$$\theta = (M + \Delta M) \cdot \frac{360}{36} \tag{33}$$

The offset larger than 0.5 will be autocratically rejected.

Assigning an orientation to each feature descriptor is to gain the invariance to image rotation by indicating the relations between the representation and the feature itself. Gradient is introduced with the purpose of avoiding the effects of the illumination changes. The patch to collect the gradient will then steer to the main direction to generate the traditional SIFT descriptor [14]. The image rotation can also be equalized as a special case of affine transformation in the form of:

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{34}$$

Combining the affine transformation, the total transformation can be synthesized as:

$$A' = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot A \tag{35}$$

By applying the gradient relocation depicted in Fig. 6, a square patch of gradient can be obtained (Fig. 7).

As depicted in Fig. 8, affine descriptor can then be generated. It has an exact same form of SIFT descriptor and can be directly applied in a lot of pre-SIFT applications

5 Camera Matrix Calculation Via the Image-Image Feature Correspondences

With the matched correspondences, we can then apply the DLT method to calculate the camera matrix to define the 2D image to 3D models position mapping (Figs. 8, 9, 10).

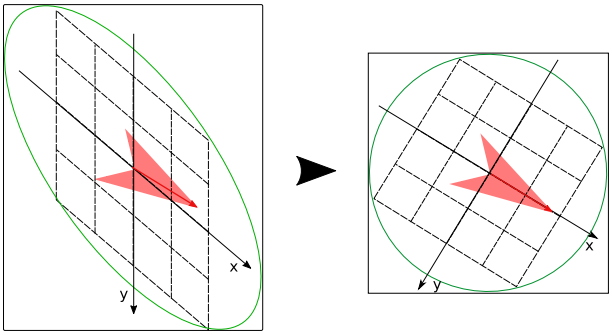

Fig. 7 Gradient relocation according to the synthesized transformation matrix. After the relocation, we will obtain a set of gradients eliminating the affine transformation and pointing to the main direction

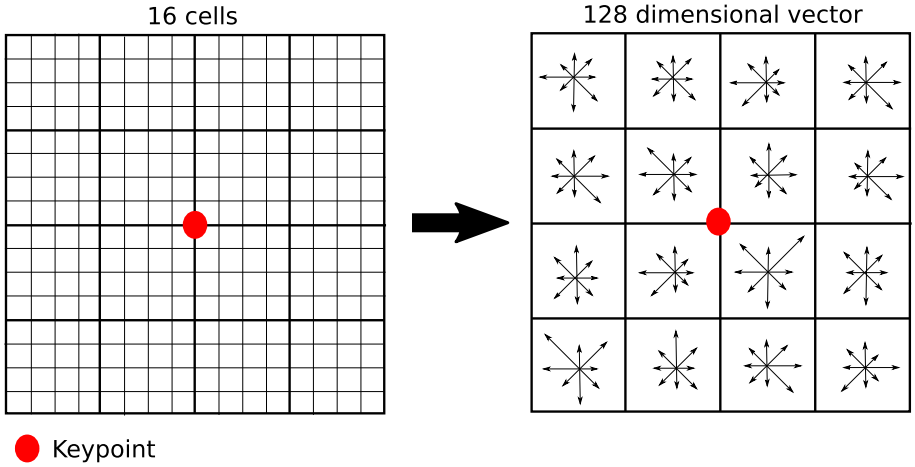

Fig. 8 Borrowing the idea of SIFT descriptor, affine descriptor can be generated from the relocated gradient

The mapping from the coordinates of a 3D point P to the 2D image coordinates of the point's projection onto the image plane, according to the pinhole camera model is given by

$$\begin{pmatrix} y_1 \\ y_2 \end{pmatrix} = \frac{f}{x_3} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \tag{36}$$

where (x_1, x_2, x_3) are the 3D coordinates of P relative to a camera centered coordinate system, (y_1, y_2) are the resulting image coordinates, and f is the camera's focal length for which we assume $f > 0$. Furthermore, we also assume that $x3 > 0$. To derive the camera matrix this expression is rewritten in terms of homogeneous coordinates. Instead of the 2D vector (y_1, y_2) we consider the projective element (a 3D vector)$\mathbf{y} = (y_1, y_2, 1)$ and instead of equality we consider equality up to scaling by a non-zero number, denoted $\sim$. First, we write the homogeneous image coordinates as expressions in the usual 3D coordinates.

$$\begin{pmatrix} y_1 \\ y_2 \\ 1 \end{pmatrix} = \frac{f}{x_3} \begin{pmatrix} x_1 \\ x_2 \\ \frac{x_3}{f} \end{pmatrix} \sim \begin{pmatrix} x_1 \\ x_2 \\ \frac{x_3}{f} \end{pmatrix} \tag{37}$$

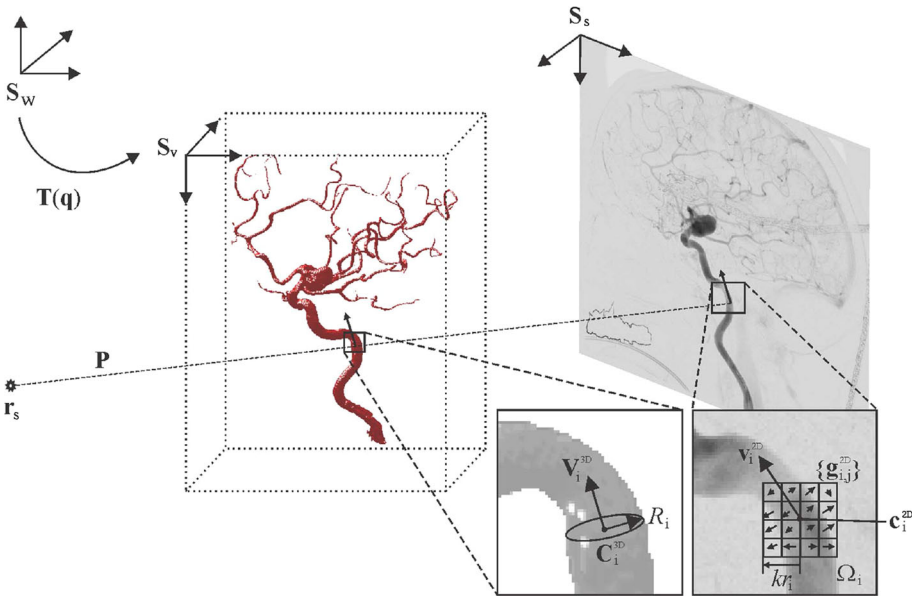

Fig. 9 A typical example of our proposed method. We will compare the performance our proposed method with some other most frequently applied algorithm on this special case [15]

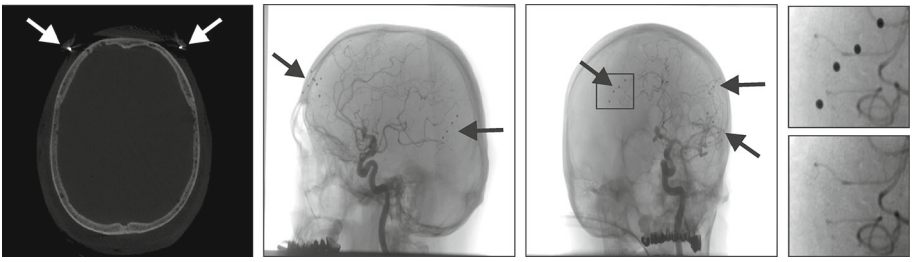

Fig. 10 Typical 3D and 2D images with fiducial markers: a CBCT axial slice (**a**), a 2D-MAX LAT (**b**) and 2D-MAX AP (**c**) images. The arrows indicate the locations of the fiducial markers [15]

Finally, also the 3D coordinates are expressed in a homogeneous representation $\mathbf{x}$ and this is how the camera matrix appears:

$$\begin{pmatrix} y_1 \\ y_2 \\ 1 \end{pmatrix} \sim \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ 1 \end{pmatrix} \tag{38}$$

Or,

$$\mathbf{y} \sim \mathbf{C}\,\mathbf{x} \tag{39}$$

where $\mathbf{C}$ is the camera matrix, which here is given by

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix} \tag{40}$$

and the corresponding camera matrix now becomes

$$\mathbf{C} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{f} & 0 \end{pmatrix} \sim \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{41}$$

The last step is a consequence of **C** itself being a projective element.

The camera matrix derived here may appear trivial in the sense that it contains very few non-zero elements. This depends to a large extent on the particular coordinate systems which have been chosen for the 3D and 2D points. In practice, however, other forms of camera matrices are common, as will be shown below.

The camera matrix **C** derived in the previous section has a null space which is spanned by the vector

$$\mathbf{n} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} \tag{42}$$

This is also the homogeneous representation of the 3D point which has coordinates (0,0,0), that is, the "camera center" (aka the entrance pupil; the position of the pinhole of a pinhole camera) is at O. This means that the camera center (and only this point) cannot be mapped to a point in the image plane by the camera (or equivalently, it maps to all points on the image as every ray on the image goes through this point).

For any other 3D point with $x_3 = 0$, the result $\mathbf{y} \sim \mathbf{C}$ is well-defined and has the form $\mathbf{y} = (y_1\, y_2\, 0)^\top$. This corresponds to a point at infinity in the projective image plane (even though, if the image plane is taken to be a Euclidean plane, no corresponding intersection point exists).

In this way, can we achieve the 2D images registered to the 3D stereo visual based models.

6 Experiment

Detecting the features across the images of different perspective is the fundamental for acquiring accurate depth information. An accurate depth information is also the fundamental for an accurate 3D to 2D image model registration. Based on this situation, we have listed our feature matching upon the images of different perspectives for 3D-2D registration in the Table 1.

In the Table, we present the performance of our proposed algorithm comparing with some most widely applied feature matching algorithms in the 3D-2D image registration, including SIFT, SURF and ALP. By comparison, our proposed method outperforms agains other feature matching algorithms specially on the 2D registration images. General speaking, our proposed 3D-2D registration method is largely depending on an accurate feature matching algorithm which can uttermost detect the potential correlations between different perspectives of images (Tables 2, 3).

The clinical image database was used to quantitatively evaluate the performances of our proposed AIFD based method comparing with three state-of-the-art 3D-2D registration methods Selection of the state-of- the-art methods was limited to methods that are well established in the field of 3D-2D registrations, and that are capable of registering a 3D image either to one 2D view or to multiple 2D views simultaneously. There were about 14,000 centerline points per 2D image for which the distance transform was precomputed so as to speed

Table 1 The average matching performance upon the images from the different perspectives (dataset *QUBR*)

Skewing	SIFT		ALP		SURF		Proposed AIFD	
	Number	Ratio	Number	Ratio	Number	Ratio	Number	Ratio
1	423.345	0.305	454.521	0.408	371.667	0.380	483.479	0.623
2	93.243	0.058	102.356	0.095	89.032	0.047	436.138	0.577
3	81.543	0.052	97.456	0.085	72.546	0.038	381.59	0.496
4	24.456	0.017	37.056	0.031	22.453	0.015	328.897	0.431
5	16.657	0.010	21.802	0.0155	15.234	0.00967	289.805	0.378
6	13.234	0.0078	18.307	0.0135	12.342	0.00503	235.845	0.322
7	5.213	0.0017	6.234	0.0029	3.434	0.00093	183.395	0.289
8	1.367	0.0013	2.921	0.00192	1.412	0.00107	157.281	0.228
9	0.572	0.0003	0.560	0.0005	0.442	0.0003	127.362	0.220

The row of skewing level refers to the level of view point difference

Table 2 Comparison of our proposed method and some state-of-the-art algorithms

View	Method	MEAN ± STD		SR (%)		CR (mm)		Time (s)
		DSA	MAX	DSA	MAX	DSA	MAX	
LAT	MIP-MI	0.32 ± 0.21	0.56 ± 0.53	72.32	34.23	4	3	76.3
	ICP	0.44 ± 0.23		42.02		1		1.1
	BGB	0.40 ± 0.37	0.41 ± 0.36	52.38	48.43	3	2	13.4
	MGP	0.61 ± 0.37	0.63 ± 0.39	73.23	69.98	5	3	0.9
	MGP+BGB	0.26 ± 0.23	0.29 ± 0.27	73.21	72.21	5	3	12.8
	AIFD	0.62 ± 0.25	0.58 ± 0.31	70.31	68.23	5	4	5.7
AP	MIP-MI	0.27 ± 0.32	0.68 ± 0.45	91.78	32.87	9	3	65.2
	ICP	0.32 ± 0.25		72.48		1		0.4
	BGB	0.32 ± 0.35	0.44 ± 0.33	58.32	52.13	3	5	13.8
	MGP	0.53 ± 0.27	0.63 ± 0.33	92.43	85.68	10	9	0.9
	MGP+BGB	0.28 ± 0.17	0.39 ± 0.27	95.45	85.3	11	8	10.5
	AIFD	0.45 ± 0.33	0.56 ± 0.21	72.34	63.21	4	4	6.8

Tested on dataset I and II. Mean and std of mtre values of successful registrations, success rates (sr), capture ranges (cr) and mean execution times averaged over all clinical datasets

up the nearest neighbor search to the projected 3D centerline points. In the BGB method the 3D intensity gradients were using the Canny edge detector, which resulted in about 17,000 edge points. The 2D intensity gradients were computed by the central difference kernel.

Parameters of the state-of-the-art 3D-2D registration methods were experimentally set to obtain the best registration performances on the clinical image dataset. For MIP-MI method, the sampling step along the projection rays was 0.375 mm and the intensities were discretized in 64 bins to compute the MI histograms. The ICP had no user-controlled parameters, while in the BGB method the sensitivity of the angle weighting function, was set to $n = 4$.

Table 3 Comparison of our proposed method and some state-of-the-art algorithms

View	Method	MEAN ± STD		SR (%)		CR(mm)		Time (s)
		DSA	MAX	DSA	MAX	DSA	MAX	
LAT	MIP-MI	0.23 ± 0.22	0.57 ± 0.46	67.32	32.12	3	3	116.3
	ICP	0.48 ± 0.33		41.57		2		1.1
	BGB	0.38 ± 0.32	0.38 ± 0.31	51.32	42.41	2	2	11.4
	MGP	0.61 ± 0.37	0.63 ± 0.39	73.23	69.98	5	2	1.8
	MGP+BGB	0.24 ± 0.17	0.26 ± 0.21	72.29	71.87	4	2	18.7
	AIFD	0.45 ± 0.34	0.23 ± 0.46	73.21	61.32	4	3	8.2

Tested on dataset III. Mean and std of mtre values of successful registrations, success rates (sr), capture ranges (cr) and mean execution times averaged over all clinical datasets

7 Conclusion

In this paper, we presented a novel method for 3D-2D rigid registration based on our previous proposed feature matching algorithm AIFD, which it more able to detect the correspondences across the images of different perspectives. The main advantage of the proposed method is it is more robust to viewpoint difference, resulting a less number of snapshot around. By experiment, it can be proven that our proposed method performs best among all the most applied registration method, and the overall execution time is also quite fast.

Translation of any 3D-2D registration method into clinical practice requires extensive and rigorous evaluations on real-patient image databases. Therefore, we acquired a clinical image database representative of cerebral-EIGI and established a highly accurate gold standard registration that enables objective quantitative evaluation of 3D-2D rigid registration methods. The quantitative and comparative evaluation of three state-of-the-art methods showed that the performance of the proposed method best met the demands of cerebral EIGI.

References

1. Datta R, Joshi D, Li J, Wang JZ (2008) Image retrieval: ideas, influences, and trends of the new age. ACM Comput Surv 40(2):5
2. Du S, Guo Y, Sanroma G, Ni D, Wu G, Shen D (2015) Building dynamic population graph for accurate correspondence detection. Medical Image Analysis
3. Du S, Liu J, Zhang C, Zhu J, Li K (2015) Probability iterative closest point algorithm for m-d point set registration with noise. Neurocomputing 157:187–198
4. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 24(6):381–395
5. Florack L, Maas R, Niessen W (1999) Pseudo-linear scale-space theory. Int J Comput Vision 31(2–3):247–259
6. Förstner W, Gülch E (1987) A fast operator for detection and precise location of distinct points, corners and centres of circular features. In: Proceeding ISPRS intercommission conference on fast processing of photogrammetric data, pp. 281–305

7. Gao Y, Ji R, Cui P, Dai Q, Hua G (2014) Hyperspectral image classification through bilayer graph-based learning. IEEE Trans Image Process 23(7):2769–2778
8. Gao Y, Ji R, Liu W, Dai Q, Hua G (2014) Weakly supervised visual dictionary learning by harnessing image attributes. IEEE Trans Image Process 23(12):5400–5411
9. Gao Y, Wang M, Tao D, Ji R, Dai Q (2012) 3-d object retrieval and recognition with hypergraph analysis. IEEE Trans Image Process 21(9):4290–4303
10. Gao Y, Wang M, Zha ZJ, Shen J, Li X, Wu X (2013) Visual-textual joint relevance learning for tag-based social image search. IEEE Trans Image Process 22(1):363–376
11. Gonzalez RC, Woods RE, Eddins SL (2004) Digital image processing using matlab. Pearson Prentice Hall, Upper Saddle River
12. Lindeberg T (1992) Scale-space behaviour of local extrema and blobs. J Math Imaging Vis 1(1):65–99
13. Lindeberg T (2013) Generalized axiomatic scale-space theory. Adv Imaging Electron Phys 178:1
14. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
15. Mitrovic U, Špiclin Ž, Likar B, Pernuš F (2013) 3D–2D registration of cerebral angiograms: a method and evaluation on clinical images. IEEE Trans. Med. Imaging 32(8):1550–1563
16. Paschalakis S, Francini G (2014) Test model 12: compact descriptors for visual search. testmodel ISO/IEC JTC1/SC29/WG11/N14961, MPEG, Strasbourg, France
17. Tuytelaars T, Mikolajczyk K (2008) Local invariant feature detectors: a survey. Found Trends Comput Gr Vis 3(3):177–280
18. Zhao B, Lepsoy S, Magli E (2015) Affine scale space for viewpoint invariant keypoint detection. In: Multimedia Signal Processing (MMSP), 2015 IEEE 17th International Workshop on. pp 1–6