CCSE Computer and Information Science
www.ccsenet.org/journal.html
Vol. 2, No. 4
November 2009

Radial Projection Fourier Transform and its Application for Scene Matching with Rotation Invariance

Lang Su (Corresponding author)
College of Mechatronic Engineering and Automation, National University of Defense Technology
De Ya Road, Changsha 410073, China
E-mail: sulang00@139.com

Zheng Gao
College of Mechatronic Engineering and Automation, National University of Defense Technology
De Ya Road, Changsha 410073, China
E-mail: gaozheng1605@163.com

Abstract

Scene matching is used for image registration in many fields. There are usually translation and an arbitrary unknown rotation angle between reference and template images. The corresponding scene matching algorithm costs far more computing time than that with small rotation angle and translation. Conceptions of generalized vector image, gray-scale image rotation transformation, gray-scale image point transformation, nature of shift invariance and rotation invariance are proposed to form the foundation of this paper. Radial Projection Fourier Transform (RPFT) is proposed and its rotation invariance is formal proved in this paper. It is applied to the Algorithm of Scene Matching with Rotation Invariance (ASMRI).

Calculation on reference and template images can be done separately. Some works can be done before the template images are required. This can improve the matching speed at the cost of more memory.

A program to implement the proposed RPFT and ASMRI based on RPFT is coded by means of Visual C++ 6.0. The results prove that ASMRI based on RPFT is not only rotation invariant, but also more accurate and faster than traditional methods. The program can be carried out with hardware.

Keywords: Radial Projection Fourier Transform, Scene Matching, Rotation Invariance, Discrete Fourier Transform, Generalized Vector Image, Gray-scale Image Point Transform

1. Introduction

Scene matching refers to an image processing technology which can find corresponding region of template images in reference images or find the correlation between them (Zhao, Fengwei, 2002, P.110-113). The images are of the same scene taken at different times, from different perspectives, or by different sensors (Barbara Zitova, 2003, P.977–1000).

In scene matching area, image matching with translation and arbitrary unknown rotation angle is fundamental and important. There have been many researches on rotation invariant image matching, such as Mean Square Difference (MSD) matching algorithm (Pang S N, 2004, P.519-527) based on the minimum average square difference, Normalized Product (NPROD) correlation matching algorithm (Su, kang, 1997, P.1-7) based on regional gray association, etc. However, those methods can perform well only when reference and template images differ in a small rotation angle. To have a better effect for scene matching with rotation of arbitrary unknown angle, Farhan Ullah (Ullah F, 2005, P.201-209) presented orientation codes for rotation invariant template matching, but the method costs much computing time which may leads to less practicability; Wang Jingdong (Wang, Jingdong, 2005, P.6-10) presented a circular projection matching algorithm, but the correct matching probability of this method is not high enough for practice; Sun Bojiao (Sun, Bojiao, 2008, P43-48) presented a matching method based on normalized cross-correlation, but the selection of step has great effect on image registration.

Conceptions of generalized vector image, gray-scale image rotation transformation, gray-scale image point transformation, nature of shift invariance and rotation invariance etc. are proposed to form the foundation of this paper.

Radial Projection Fourier Transform (RPFT) is proposed and its rotation invariance is formal proved in this paper. It is applied to the Algorithm of Scene Matching with Rotation Invariance (ASMRI). It involves eight steps as follows:

step1: extract the edge of original reference image to form reference edge image;

step2: extract Subsets of Suitable Matching Points (SSMP) from reference image which includes original reference image and reference edge image;

step3: calculate Radial Projection Fourier Transform (RPFT) of each point in SSMP of reference image and save it into memory;

step4: extract the edge of original template images to form template edge images;

step5: extract SSMP from template images which includes original template image and template edge image;

step6: calculate RPFT of each point in SSMP of template image;

step7: match the result of MDFT of each RPV utilizing proper comparability measurement to confirm the corresponding points;

step8: calculate the rotation angle between reference and template images.

In most scene matching, the reference image is obtained earlier. Therefore, the computations on reference image can be fully implemented offline. In this paper, the result of the computations on reference image was saved into memory. Then the matching process can be quickly done after reading in the saved data when the template images are acquired, which may greatly improve the matching speed.

This paper is organized as follows. Section 2 describes generalized vector image in detail. Section 3 describes gray-scale image point transformation and its nature of shift invariance and rotate invariance. In section 4, RPFT is proposed and its nature of rotation invariance is formal proved. In section 5, ASMRI based on RPFT is presented. In section 6, experiments of ASMRI based on RPFT are presented and compared with that of tradition methods. Finally, conclusions are drawn in section 7.

2. Generalized Vector Image

In order to interpret RPFT better, the formal definitions of concepts of gray-scale image, gray-scale image shifting transformation, shifting image, gray-scale image rotation transformation, set of rotation transformations of gray-scale image, rotating image, rotation corresponding point, generalized image, N-dimensional generalized vector image, vector image shifting transformation, vector shifted image, vector image rotation transformation and vector rotated image are provided first basing on the describing method of Mathematical Logic and Set Theory (Geng, Suyun, 2002, Shi, Chunyi, 2000)

Definition 1: A *gray-scale image* is a map from R^2 **to** R, where R is the real number domain. In other words, a gray-scale image is a real function of two variables, it can be noted as $f(x,y)$ or $f: R^2 \to R$. $f(x_0, y_0)$ means the grey value of the pixel with the coordinate of (x_0, y_0). The set of all gray-scale images is noted as GI.

Definition 2: If a map from GI to GI satisfies Proposition 1, the map can be referred to as a *gray-scale image shift transformation* with the shift of (dx, dy). It can be noted as $SHIFT_{dx,dy}$, i.e. $SHIFT_{dx,dy} : GI \to GI$.

Proposition 1: If $g(x,y) = SHIFT_{dx,dy}[f(x,y)]$, then

$$\forall i, j \in R : g(i,j) = f(i+dx, j+dy). \tag{1}$$

Definition 3: If $f(x,y)$ and $g(x,y)$ satisfies

$$g(x,y) = SHIFT_{dx,dy}[f(x,y)], \tag{2}$$

$g(x,y)$ can be noted as a *shifting image* of $f(x,y)$ with the shift of (dx,dy). That is to say, $g(x,y)$ is the result of the shift of (dx,dy) on $f(x,y)$.

Definition 4: If a map from GI to GI satisfies Proposition 2 or 3, the map can be noted as a *gray-scale image rotation transformation* with the rotation of (x_0, y_0) and the angle of α .It can be noted as $ROT_{x_0,y_0,\alpha}$.

Proposition 2: If $g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)]$, then $\forall x_1, x_2, y_1, y_2 \in R$,

$$\begin{cases} x_2 - x_0 = (x_1 - x_0)\cos\alpha - (y_1 - y_0)\sin\alpha \\ y_2 - y_0 = (x_1 - x_0)\sin\alpha + (y_1 - y_0)\cos\alpha \end{cases} \to g(x_2, y_2) = f(x_1, y_1). \tag{3}$$

Proposition 3: If $g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)]$, then $\forall \rho_1,\rho_2,\theta_1,\theta_2 \in R$,

$$\begin{cases} \rho_2 = \rho_1 \\ \theta_2 = \theta_1 + \alpha \end{cases} \rightarrow g(\rho_2 \cos\theta_2 + x_0, \rho_2 \sin\theta_2 + y_0) = f(\rho_1 \cos\theta_1 + x_0, \rho_1 \sin\theta_1 + y_0) . \quad (4)$$

Definition 5: A *set of rotation transformations of gray-scale image* is defined as:

$$ROTSet = \bigcup_{x\in R, y\in R, \alpha\in R} \{ROT_{x,y,\alpha}\} . \quad (5)$$

In other words, set of rotation transformations of gray-scale image is a set of gray-scale image rotation transformations with the arbitrary rotation of (x,y) or with any rotate angle of α .

Definition 6: If $f(x,y)$ and $g(x,y)$ satisfies

$$g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)] , \quad (6)$$

$g(x,y)$ can be noted as *rotating image* of $f(x,y)$ with the arbitrary rotation of (x_0,y_0) and the rotate angle of α .

Definition 7: If (x_2,y_2) and (x_1,y_1) satisfies

$$\begin{cases} x_2 - x_0 = (x_1 - x_0)\cos\alpha - (y_1 - y_0)\sin\alpha \\ y_2 - y_0 = (x_1 - x_0)\sin\alpha + (y_1 - y_0)\cos\alpha \end{cases} , \quad (7)$$

point (x_2,y_2) can be noted as *rotation corresponding point* of point (x_1,y_1) based on $ROT_{x_0,y_0,\alpha}$.

Definition 8: A *generalized image* is a map from R^2 to S , in which S is a set. Most transformation of image can be described by the conception of generalized image. The form of S can be different in different applications.

Definition 9: N - *dimensional* *generalized vector image* is a kind of generalized image, "generalized vector image" for short. It is a map from R^2 to V_N , where V_N is N-dimensional Euclidean Vector Space, which is a set of all N-dimensional vectors. For example, a gray-scale image is a one-dimensional generalized vector image; a color image is a three-dimensional generalized vector image; Radial Projection Vector Field expatiated in this paper is a 512-dimensional generalized vector image. That is to say, N-dimensional generalized vector image is a vector function of two variables. It can be noted as $\boldsymbol{f}(x,y)$, i.e. $\boldsymbol{f} : R^2 \rightarrow V_N$.

Definition 10: If a map from VI_N to VI_N satisfies Proposition 4, the map can be called a *vector image shifting transformation* with the shift of (dx,dy) .It can also be noted as $\boldsymbol{SHIFT}_{dx,dy}$.

Proposition 4: If $\boldsymbol{g}(x,y) = \boldsymbol{SHIFT}_{dx,dy}[\boldsymbol{f}(x,y)]$, then

$$\forall i,j \in R : \boldsymbol{g}(i,j) = \boldsymbol{f}(i+dx, j+dy) . \quad (8)$$

Definition 11: If N-dimensional generalized vector image $\boldsymbol{f}(x,y)$ and $\boldsymbol{g}(x,y)$ satisfies

$$\boldsymbol{g}(x,y) = \boldsymbol{SHIFT}_{dx,dy}[\boldsymbol{f}(x,y)], \quad (9)$$

$\boldsymbol{g}(x,y)$ can be noted as *vector shifted image* of $\boldsymbol{f}(x,y)$ with the shift of (dx,dy) .

Definition 12: If a map from VI_N to VI_N satisfies Proposition 5 or 6, the map can be noted as a *vector image rotation transformation* with the shift of (dx,dy) .It can be noted as $\boldsymbol{ROT}_{x_0,y_0,\alpha}[\boldsymbol{f}(x,y)]$.

Proposition 5: If $\boldsymbol{g}(x,y) = \boldsymbol{ROT}_{x_0,y_0,\alpha}[\boldsymbol{f}(x,y)]$, then $\forall x_1,x_2,y_1,y_2 \in R$,

$$\begin{cases} x_2 - x_0 = (x_1 - x_0)\cos\alpha - (y_1 - y_0)\sin\alpha \\ y_2 - y_0 = (x_1 - x_0)\sin\alpha + (y_1 - y_0)\cos\alpha \end{cases} \rightarrow \boldsymbol{g}(x_2,y_2) = \boldsymbol{f}(x_1,y_1) . \quad (10)$$

Proposition 6: If $\boldsymbol{g}(x,y) = \boldsymbol{ROT}_{x_0,y_0,\alpha}[\boldsymbol{f}(x,y)]$, then $\forall \rho_1,\rho_2,\theta_1,\theta_2 \in R$,

$$\begin{cases} \rho_2 = \rho_1 \\ \theta_2 = \theta_1 + \alpha \end{cases} \rightarrow \boldsymbol{g}(\rho_2 \cos\theta_2 + x_0, \rho_2 \sin\theta_2 + y_0) = \boldsymbol{f}(\rho_1 \cos\theta_1 + x_0, \rho_1 \sin\theta_1 + y_0) \tag{11}$$

Definition 13: If N-dimensional generalized vector image $\boldsymbol{f}(x, y)$ and $\boldsymbol{g}(x, y)$ satisfies

$$\boldsymbol{g}(x, y) = \boldsymbol{ROT}_{x_0, y_0, \alpha}[\boldsymbol{f}(x, y)], \tag{12}$$

$\boldsymbol{g}(x, y)$ can be called *vector rotated image* of $\boldsymbol{f}(x, y)$ with the rotation of (x_0, y_0) and rotate angle of α .

3. Gray-scale image point transformation

The formal concept of gray-scale image point transformations is proposed first, and then its transformation invariance and rotation invariance is defined.

Definition 14: A *gray-scale image point transformation* is a map from GI to VI_N .It can be noted as $GIPT$, i.e. $GIPT : GI \rightarrow VI_N$.

Definition 15: A *set of gray-scale image point transformations* is a set of all gray-scale image point transformations, it can be noted as $GIPTSet$.

Definition 16: A gray-scale image point transformation possesses the nature of *shift invariance*, if the shift of input image leads to the same shift in output image. That is to say,

if $GIPT \in GIPTSet$ is shift invariant, and $\forall f(x, y) \in GI, g(x, y) \in GI, dx \in R, dy \in R$,

$$g(x, y) = SHIFT_{dx,dy}[f(x, y)] \rightarrow GIPT[g(x, y)] = \boldsymbol{SHIFT}_{dx,dy}[GIPT[f(x, y)]] . \tag{13}$$

Definition 17: A gray-scale image point transformation possesses the nature of *rotation invariance*, if the rotation of input image leads to the same rotation in output image. That is to say,

if $GIPT \in GIPTSet$ is rotation invariant , and $\forall f(x, y) \in GI, g(x, y) \in GI, dx \in R, dy \in R$,

$$g(x, y) = ROT_{x_0, y_0, \alpha}[f(x, y)] \rightarrow GIPT[g(x, y)] = \boldsymbol{ROT}_{x_0, y_0, \alpha}[GIPT[f(x, y)]] . \tag{14}$$

4. Radial Projection Fourier Transform

4.1 The definition of RPFT

The definition of Radial Projection Transform, Radial Projection Vector, and Radial Projection Vector Field is given first, and then Radial Projection Fourier Transform is proposed based on DFT and modulus operator.

With regard to N-dimensional vector $\boldsymbol{x}$, $\boldsymbol{x} = \{\boldsymbol{x}[n]\}_{n=0}^{N-1}$, and the *n*th element is noted as $\boldsymbol{x}[n]$.

Definition 18: *Radial Projection Vector* (RPV) is a kind of N-dimensional vector. If the rectangular coordinate of point A in grey-level image $f(x, y)$ is (x_0, y_0) , then RPV of point A in grey-level image $f(x, y)$ can be noted as $\boldsymbol{RPV}_{f,A}$. $\boldsymbol{RPV}_{f,A}[n]$ is calculated as formula 15.

$$\boldsymbol{RPV}_{f,A}[n] = \sum_{\rho=1}^{\text{radius}} f(\rho\cos\theta + x_0, \rho\sin\theta + y_0) , n = 0, 1, ..., N-1 \tag{15}$$

Where $\theta = \dfrac{2\pi}{N} n$. *radius* is the radius of the rotundity neighborhood of point A .

That is to say, RPV of a pixel point is to project grey-level value of its rotundity neighborhood. According to the method in formula 15, grey-level value of different angle is projected to vector of specifically dimensions.

As is shown in Fig1, (a) is grey-level image $f(x, y)$, in which, a specified point A is pointed by a white line; (b) is Schematic diagram of Radial Projection of point A ; (c) is $\boldsymbol{RPV}_{f,A}$.

Definition 19:

Radial Projection Transform (RPT) is a kind of $GIPT$, it belongs to $GIPTSet$.

$$RPT : GI \rightarrow VI_N . \tag{16}$$

RPT: Calculate corresponding RPVs all points in gray-level image.

Radial Projection Vector Field (RPVF) is the result of RPF on each point of gray-level image $f(x, y)$ .It can be noted as RPT_f, or $RPT_f = RPT(f)$. It is a kind of N - dimensional generalized vector image. To any pixel point A in image $f(x,y)$,

$$RPT_f(A) = \boldsymbol{RPV}_{f,A}. \tag{17}$$

It is easy to prove that RPF is shift invariant, but not rotation invariant.

Definition 20:

Modulus Discrete Fourier Transform of vector **(**MDFT) is a map from N-dimensional Euclidean Space to N-dimensional Euclidean Space, i.e. $MDFT: R^N \to R^N$. (Alanv. Oppenheim, 1998, Guan, Zhizhong, 2004)

To N-dimensional vector $\boldsymbol{x}$, if $MDFT(\boldsymbol{x})=\boldsymbol{XM}$, then

$$\boldsymbol{XM}[k] = \left|\frac{1}{N}\sum_{n=0}^{N-1} x[n]\ W_N^{kn}\right|, \qquad k = 0,1,2........N-1 \tag{18}$$

where $W_N = \exp[-j2\pi / N]$.

Definition 21:

Point-by-point Modulus Discrete Fourier Transform of vector (PMDFT) is the result of calculating MDFT of each point in a generalized vector image, it is a map from VI_N to VI_N. It can be noted as $PMDFT: VI_N \to VI_N$.

If $\boldsymbol{g} = PMDFT(\boldsymbol{f})$, $\forall(a,b) \in R^2$: $\boldsymbol{g}(a,b) = MDFT(\boldsymbol{f}(a,b))$.

Definition 22:

Radial Projection Fourier Transform (RPFT) is the composite map of RPT and PMDFT.

$$\underbrace{GI \xrightarrow{RPT} VI_N \xrightarrow{PMDFT}}_{RPFT} VI_N \tag{19}$$

4.2 Proof of rotation invariance of RPFT

Definition 23:

Circular shift (Alanv. Oppenheim, 1998) on N-dimensional vector $\boldsymbol{x}$ with the shift of m is still a N-dimensional vector, it can be noted as $\tilde{\boldsymbol{x}}_m$, and

$$\forall n \in [0, N-1], \tilde{\boldsymbol{x}}_m[n] = \boldsymbol{x}[(n-m)(\text{mod } N)]. \tag{20}$$

Where m,n is integer, i.e. $m,n \in Z$ .That is to say, there is a circular shift with the shift of m between $\tilde{\boldsymbol{x}}_m$ and $\boldsymbol{x}$, i.e. $\tilde{\boldsymbol{x}}_m$ is the *circular shift vector* of $\boldsymbol{x}$ with the shift of m.

Lemma 1:

If $g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)]$, where $\alpha = \dfrac{2\pi}{N}m, m \in Z$, point A and B is rotation corresponding point on $ROT_{x_0,y_0,\alpha}$, $\boldsymbol{RPV}_{f,A}$ and $\boldsymbol{RPV}_{g,B}$ is corresponding RPV, then there is a circular shift with the shift of m between $\boldsymbol{RPV}_{f,A}$ and $\boldsymbol{RPV}_{g,B}$.

It can be formally described as:

$$g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)] \to (\boldsymbol{x} = \boldsymbol{RPV}_{f,A} \to \tilde{\boldsymbol{x}}_m = \boldsymbol{RPV}_{g,B}) \tag{21}$$

Proof:

According to Definition 6 about rotating image, if $g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)]$,

with regard to corresponding rotation points $A(\rho_1,\theta_1)$ and $B(\rho_2,\theta_2)$,

$$\begin{cases} \rho_2 = \rho_1 \\ \theta_2 = \theta_1 + \alpha \end{cases} \to g(\rho_2\cos\theta_2 + x_0, \rho_2\sin\theta_2 + y_0) = f(\rho_1\cos\theta_1 + x_0, \rho_1\sin\theta_1 + y_0) \tag{22}$$

According to Definition 15 about RPV, $\forall n \in [0, N-1]$,

$$\boldsymbol{RPV}_{f,A}[n] = \sum_{\rho=1}^{\text{radius}} f(\rho\cos\theta + x_0, \rho\sin\theta + y_0)\,,\ \boldsymbol{RPV}_{g,B}[n] = \sum_{\rho=1}^{\text{radius}} g(\rho\cos\theta + x_0, \rho\sin\theta + y_0)\,,$$

where $\theta = \dfrac{2\pi}{N}n$.

So, there is a circular shift between $\boldsymbol{RPV}_{f,A}$ and $\boldsymbol{RPV}_{g,B}$, and the shift equals to $\dfrac{\alpha N}{2\pi}$.

In conclusion, Formula 19 is proved.

Lemma 2:

If there is a circular shift between N-dimensional vector $\boldsymbol{y}$ and $\boldsymbol{x}$, then the result of MDFT of $\boldsymbol{y}$ equal to that of $\boldsymbol{x}$ .It can be formal described as: if $\boldsymbol{XM} = MDFT(\boldsymbol{x})$, $\boldsymbol{YM} = MDFT(\boldsymbol{y})$,

$$\forall m \in [0, N-1],\ \boldsymbol{y} = \tilde{\boldsymbol{x}}_m \to \forall k \in [0, N-1], \boldsymbol{YM}[k] = \boldsymbol{XM}[k]\,. \tag{23}$$

Proof:

According to the property of circular shift (Alanv. Oppenheim, 1998),

$\&\ \forall n \in [0, N-1],\ \boldsymbol{y}[n] = \boldsymbol{x}[(n-m)(\text{mod } N)]$.

$\forall k \in [0, N-1]$,

$$\begin{aligned}
\boldsymbol{YM}[k] &= \left|\sum_{n=0}^{N-1} \boldsymbol{y}[n]W^{kn}\right| = \left|\sum_{n=0}^{N-1} \boldsymbol{x}[(n-m)(\text{mod } N)]W^{kn}\right| \\
&= \left|\sum_{n=0}^{m-1} \boldsymbol{x}[(n-m)(\text{mod } N)]W^{kn}\right| + \left|\sum_{n=m}^{N-1} \boldsymbol{x}[(n-m)(\text{mod } N)]W^{kn}\right| \\
&= \left|\sum_{n=0}^{m-1} \boldsymbol{x}[N+n-m]W^{kn}\right| + \left|\sum_{n=m}^{N-1} \boldsymbol{x}[n-m]W^{kn}\right| \\
&= \left|\sum_{l=N-m}^{N-1} \boldsymbol{x}[l]W^{k(m+l)}\right| + \left|\sum_{l=0}^{N-1-m} \boldsymbol{x}[l]W^{k(m+l)}\right| = \left|\sum_{l=0}^{N-1} \boldsymbol{x}[l]W^{k(m+l)}\right| \\
&= \left|W_N^{mk}\sum_{l=0}^{N-1} \boldsymbol{x}[l]W^{kl}\right| = \left|\sum_{l=0}^{N-1} \boldsymbol{x}[l]W^{kl}\right| = \boldsymbol{XM}[k]\,.
\end{aligned}$$

$$\boldsymbol{YM}[k] = \boldsymbol{XM}[k]\,. \tag{24}$$

The MDFT of rotation corresponding points can keep invariance.

Theorem 1: Rotation Invariance of RPFT

If $g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)]$, and $\alpha = \dfrac{2\pi}{N}n$, then with regard to rotation corresponding points $A(x_1, y_1)$ and $B(x_2, y_2)$, the RPFT of them can keep invariance.

$$g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)] \to \boldsymbol{RPFT}[g(x,y)] = \boldsymbol{ROT}_{x_0,y_0,\alpha}[\boldsymbol{RPFT}[f(x,y)]] \tag{25}$$

Proof:

Combination at Definition 1-23,

According to formula 19, 20,

$g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)] \to (\boldsymbol{x} = \boldsymbol{RPV}_{f,A} \to \tilde{\boldsymbol{x}}_m = \boldsymbol{RPV}_{g,B})$.

Calculate the MDFT of RPV of point A and B .

According to formula 22, if $\boldsymbol{XM} = MDFT(\boldsymbol{x})$, $\boldsymbol{YM} = MDFT(\boldsymbol{y})$,

$\forall m \in [0, N-1],\ \boldsymbol{y} = \tilde{\boldsymbol{x}}_m \to \forall k \in [0, N-1], \boldsymbol{YM}[k] = \boldsymbol{XM}[k]$.

$\boldsymbol{RPFT}[f(x_1, y_1)] = (\boldsymbol{XM}[0], \boldsymbol{XM}[1], ..., \boldsymbol{XM}[N-1])$,

$\boldsymbol{RPFT}[g(x_2, y_2)] = (\boldsymbol{YM}[0], \boldsymbol{YM}[1], ..., \boldsymbol{YM}[N-1]),$

So, $\boldsymbol{RPFT}[g(x_2, y_2)] = \boldsymbol{RPFT}[f(x_1, y_1)].$

According to the randomicity of A and B , and $g(x, y) = ROT_{x_0, y_0, \alpha}[f(x, y)]$,

$\boldsymbol{RPFT}[g(x, y)] = \boldsymbol{ROT}_{x_0, y_0, \alpha}[\boldsymbol{RPFT}[f(x, y)]]$.

In conclusion, Theorem 1 is proved.

5. Implement of arithmetic of RPFT

Implement of arithmetic of RPFT consists of three steps.

step1: set up polar coordinates on each point of gray-scale image $f(x, y)$, and uses a reverse transform method [12] to calculate grey level in polar coordinates corresponding to Cartesian coordinates;

step2: calculate RPVF of image $f(x, y)$;

step3: calculate PMDFT of image $f(x, y)$.

Overview of this method is illustrated in Figure 2.

5.1 Inverse transform in polar coordinates

To find the correspondence between Cartesian coordinate system and polar coordinates, the general method is to transform Cartesian coordinates to polar coordinates: calculate the distance and angle (ρ, θ) mapping from the position of given pixel (x, y) .

Polar coordinates system (Wang, Qi, 2006, P.6-7):

$$\begin{cases} \rho \ = \ \sqrt{(x - x_0)^2 + (y - y_0)^2} \\ \theta = \arctan((y - y_0) / (x - x_0)) \end{cases} \tag{26}$$

where x_0, y_0 refers to the coordinates of a point(noted as K) in Cartesian coordinates which is the origin of polar coordinates.

Combination at the shortcomings of conventional methods, in this paper, reverse polar coordinates transform algorithm (Wang, Qi, 2006, P.6-7) is presented, which is able to overcome the above disadvantages.

Specific ideas are: utilizing polar coordinates of pixel (ρ, θ) to calculate its corresponding value of each axis in the Cartesian coordinates (x, y) . That is,

$$\begin{cases} x = \rho \cos\theta + x_0 \\ y = \rho \sin\theta + y_0 \end{cases} \tag{27}$$

Schematic diagram is shown in Figure 3 (Wang, Qi, 2006, P.6-7, Karl C. Walli, 2005, P.8).

Generally, location of corresponding points in Cartesian coordinates has a decimal value, it locates between four pixels. Based on the idea of interpolation (Karl C. Walli, 2005, P.8), the original Cartesian coordinate system can construct out of an "optimal equivalence" of pixels. Its concrete steps are as follows:

1) The reciprocal of distances from the four surrounding pixel points to the actual coordinates can be defined as the weight proportion of each surrounding pixel points respectively. The distances are noted as d_1, d_2, d_3, d_4 .

2) Calculate the value of $F(x, y)$ (Karl C. Walli, 2005, P.8)

$$F(x, y) = \frac{F(I(x), I(y))/d_1 + F(I(x)+1, I(y))/d_2 + F(I(x), I(y)+1)/d_3 + F(I(x)+1, I(y)+1)/d_4}{1/d_1 + 1/d_2 + 1/d_3 + 1/d_4} \tag{28}$$

where $I(x)$ refers to the maximum integer less than x .

Inverse transform in polar coordinates is shown in Figure 4. (a) shows input image and Cartesian coordinate with the origin of point K (240,300), (b) shows inverse transform in polar coordinates with the origin of K .

After these transformations, each point of polar coordinates has an “optimal” corresponding grey value in Cartesian coordinate, which can guarantee that each polar coordinate point has only a corresponding value.

5.2 Radial Projection Vector

Set up polar coordinates with the origin of arbitrary point A in image $f(x,y)$. According to Definition 18, RPV of point A in $f(x,y)$ as $\boldsymbol{RPV}_{f,A}$ is defined as:

$$\boldsymbol{RPV}_{f,A}[n] = \sum_{\rho=1}^{radius} f(\rho\cos\theta + x_0, \rho\sin\theta + y_0), n = 0,1,...,N-1 \tag{29}$$

where $\theta = \frac{2\pi}{N} n$ and $radius$ is the radius of neighborhood area of point A.

512-dimensional vector is adopted to denote the RPV of each point in this paper. Because DFT is designed to meet the precondition that points of signal N satisfies $N = 2^t$. If a RPV is the circular shift vector of another one, RPV of 360 elements can not satisfy the nature of circular shift when $N = 2^t$ after extension.

According to Lemma 1, if there is a rotation of unknown angle between reference image and template image, RPVs of rotation corresponding points differ in circular shift.

$$g(x,y) = ROT_{x_0,y_0,\alpha}[f(x,y)] \rightarrow (\boldsymbol{x} = \boldsymbol{RPV}_{f,A} \rightarrow \tilde{\boldsymbol{x}}_m = \boldsymbol{RPV}_{g,B}) \tag{30}$$

where $m = \frac{\alpha N}{2\pi}$.

As it is shown in Figure 5, (b) is an image of the size 300×250 pixels which is a subscene of an image which only differs in an unknown rotation angle with (a). (c) and (d) is respectively the RPV of rotation corresponding points A and B .As can be seen, there is circular shift between the corresponding RPV of points A and B.

5.3 Modulus Discrete Fourier Transform

The computation of DFT of N-dimensional vector is $O(N^2)$, which could not meet the needs of practical applications. The computation of FFT of N-dimensional vector is $O(N\log_2^N)$ .To speed up the implementation process, FFT is used to implement DFT in the experiment.

According to Lemma 2, if the RPVs differ in a circular shift with the shift of m, the result of MDFT maintain invariance.

As is shown in Figure 6, (a) and (b) is respectively MDFT of RPVs in Figure 5(c) and Figure 5(d).

6. ASMRI based on RPFT

According to rotation invariant of RPFT, it can be applied to ASMRI.

Offline calculation on reference images involves three steps:

step1: extract the edge of original reference image to form reference edge image;

step2: extract SSMP from original reference image and reference edge image respectively;

step3: calculate RPFT on each point from SSMP of reference image and calculate corresponding MDFT.

Calculation after acquiring template images consists of five steps:

step1: extract the edge of original template images to form template edge images;

step2: extract SSMP from template images, select several points from SSMP of original template and template edge images with adjacent coordinate for matching;

step3: calculate RPFT on each point from SSMP of template image and calculate MDFT on each point;

step4: match the result of RPFT of each selected points in template images with that of reference image utilizing proper comparability measurement to confirm the corresponding points;

step5: calculate the rotation angle between reference and template images;

Overview of those steps is illustrated in Figure 7.

6.1 Extraction of edge images

The edge of the images reflects step change of grey level of adjacent pixels. In this paper, the operator of Prewitt with orientation is adopted (Lu, Zongqi, 2006, P192). The result of this processing keeps the form of gray-level images. Templates of Prewitt with orientation are shown in Figure 8.

To each pixel, calculate the gradient of eight orientations, and choose the maximum of them as the output value.

As shown in Figure 9, (b) is the grey level edge image corresponding to (a), and (d) is the grey level edge image corresponding to (c).

6.2 Extraction of SSMP

For some points which are poor in grey level changes with surrounding pixels, the mistake matching probability may be unacceptable. To enhance the matching accuracy and reduce the matching time, SSMP are extracted from images. In this paper, method based on Harris corner extraction method (C.Harris, 1988, P.147-151) is adopted.

Extracting SSMP in template images involves four steps:

1) Filter to each pixel in the image utilizing horizontal and vertical difference operator to confirm I_x and I_y, which stands for horizontal and vertical changes, and calculate matrix $M = \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix}$;

2) Perform Gauss smoothing filter to four elements in matrix M, and update M .Where $Gauss = \exp(-\frac{x^2+y^2}{2\sigma^2})$;

3) Calculate corner value of every pixel. $Corner\ value = \frac{I_x^2 I_y^2 - (I_x I_y)^2}{I_x^2 + I_y^2}$;

4) Extract corner which satisfies "corner value is not less than 0.6 multiply the maximum corner value of all pixels" and "its corner value is the maximum of its neighboring pixels".

Extracting SSMP in reference images also involves four steps. It is the same as the method above from step 1 to step 3. Instead, step 4 is to extract corners which satisfy "corner value is not less than 0.30 multiply the maximum corner value of all pixels" and "its corner value is not less than 0.60 multiply the maximum of its neighboring pixels".

The steps of extracting SSMP are illustrated in Figure 10. (a) and (b) shows original reference image and template images, (c) and (d) shows reference edge image and template edge images, (e) and (f) shows SSMP extracted from original reference image and template images, (g) and (h) shows SSMP extracted from reference edge images and template edge images.

This can insure that SSMP from template and template edge images can find its corresponding points in reference and reference edge images.

6.3 Determination of corresponding points

After extracting SSMP and calculating their RPVs, matching can be performed to determine corresponding points.

As shown in Figure 6, the value of the both ends of MDFT is much larger than the others. The algorithm is easy to degenerate to be the algorithm of find the point of similar sum of the surrounding grey level, because for MDFT, $\boldsymbol{XM}[0] = \left| \frac{1}{N} \sum_{n=0}^{N-1} \boldsymbol{x}[n] \right|$. Thus $[\boldsymbol{RPFT}[f(x,y)]]$ and $\boldsymbol{RPFT}[g(x,y)]$ should be normalized in matching process, making each result of MDFT have equal or similar impact. $[\boldsymbol{RPFT}[f(x1,y1)]]$ and $\boldsymbol{RPFT}[g(x2,y2)]$ is respectively noted as $\boldsymbol{XM}$ and $\boldsymbol{YM}$ .Combined with the method of the similarity of multi-dimensional vector, similarity measure function (Du, Jie, 2007, P11) is defined as:

$$f = \frac{\sum_{k=0}^{n-1} (\frac{\boldsymbol{XM}[k]}{\overline{\boldsymbol{XM}[k]}} - 1) \times (\frac{\boldsymbol{YM}[k]}{\overline{\boldsymbol{YM}[k]}} - 1)}{\sqrt{\sum_{k=0}^{n-1} (\frac{\boldsymbol{XM}[k]}{\overline{\boldsymbol{XM}[k]}} - 1)^2 \sum_{k=0}^{n-1} (\frac{\boldsymbol{YM}[k]}{\overline{\boldsymbol{YM}[k]}} - 1)^2}} \tag{32}$$

where $\overline{\boldsymbol{XM}[k]} = \sum_{k=0}^{jdleft} \boldsymbol{XM}[k] \Big/ jdleft$, $\overline{\boldsymbol{YM}[k]} = \sum_{k=0}^{jdright} \boldsymbol{YM}[k] \Big/ jdright$, $jdleft, jdright$ is respectively the number of points of SSMP in template and reference images.

It is the same principle in determination of corresponding point in reference edge image and template edge images.

6.4 Determination of rotation angle

After acquiring corresponding points M and M' of reference and template images, rotation angle exists between them can be determined.

Calculate corresponding PRV of M, M' separately, noted as $RPV_{f,M}$ and $RPV_{g,M'}$, and then offset $RPV_{g,M'}$ from 0 to 511. According to the offset when relevance of both vectors is the maximum of all, angle of rotation can be determined.

Angle of rotation between reference edge and template edge images can be determined using above methods.

7. Experimental Results

A program to implement the proposed RPFT and ASMRI based on RPFT is coded by means of Visual C++6.0. In this section, three scene matching experiments are presented. For each experiment, reference images, template images, matching accuracy and matching speed are shown. Experiments are done on the PC platform with a CPU of 2.8GHz. In those experiments, *raidus* is 20. The interface of the program is shown in Figure 11.

7.1 Experiment 1: rotation between reference and template images

The first experiment demonstrates robustness of the method to scene matching with rotation between reference and template images. Rotation angle varies from 5 ° to 355 ° with interval of 5 ° during taking photos, totally 71 images. Four pairs of images are considered in this experiment as shown in Figure 12.

7.2 Experiment 2: rotation and grey change between reference and template images

The second experiment demonstrates robustness of the method to scene matching with rotation and grey change between reference and template images. Rotation angle varies from 5 ° to 355 ° with interval of 5 ° . Four pairs of images are considered in this experiment as shown in Figure 13.

7.3 Experiment 3: rotation and Gaussian noise between reference and template images

The third experiment demonstrates robustness of the method to scene matching with rotation and Gaussian noise between reference and template images. Rotation angle varies from 5 ° to 355 ° with interval of 5 ° . Four pairs of images are considered in this experiment as shown in Figure 14.

Four pairs of images are considered in those experiments as shown in Figure 12, 13, 14. (b), (d), (f), (h) is respectively one of the 71 images of corresponding images of (a), (c), (e) and (g). The experimental results are shown respectively in table 1, 2, 3.

As can be seen from Table 1, 2 and 3, matching accuracy rate of this algorithm is higher than the circular projection algorithm (Wang, Jingdong, 2005, 6-10). When the template images have a change in brightness or are interfered by Salt-Pepper noise, matching accuracy rate of circular projection algorithm is not high, while the algorithm proposed in this paper can still achieve a high matching rate.

8. Conclusion

In scene matching, there is usually a difference in rotation angle between reference and template images, which may lead to mistake matching. The algorithm proposed in this paper maintains a very high matching accuracy with arbitrary rotation angle and keep robust with rotation and gray change, rotation and Gaussian noise in template image. The analysis of the reference image have been completed earlier, together with the RPV is only operated by 1D-DFT, so the matching process needs only less computation. Combination at the different character of original image and edge image can enhance the matching accuracy.

In all, the algorithm proposed in this paper possesses the nature of rotation invariance, has a high matching rate, matching speed and strong applicability in scene matching with rotation invariance, it will surely has a wide application.

References

Alanv. Oppenheim, & Ronald W. Schafer, John R. Buck. (1998). *Discrete-time signal processing*, (2nd ed.). (Chapter 6).

An, Ru.(2005). Investigation of image matching algorithm for aircraft navigation based on corner feature. Doctor's degree paper, Nanjing University. 200505. 88.

Barbara Zitova, & Jan Flusser. (2003). Image registration methods: a survey. *Image and Vision Computing*. 21(2003). 977–1000.

C.Harris, & M. Stephens,(1988). ”A combined corner and edge detector”. Alvey Vision Conf. 147-151.

Du, Jie.(2007). Two fast image matching algorithms based on gray value. Master's degree paper, Dalian maritime university. 10-11.

Geng, Suyun, Qu, Wanlin, & Wang, Hanpin. *Tutorial of Discrete Mathmatics*. Press of PeKing University. 2002.06.

Guan, Zhizhong, Xia, Gongge, & Meng, Qiao. (2004). *Signal and linear system*. Higher Education Press, China. 114.

Karl C. Walli. (2005). Multisensor image registration utilizing the LOG filter and FWT. Master's degree paper, Rochester Institute of Technology.

Pang S N, Kim H C, & Kim D,et al. (2004). Prediction of the suitability for image-matching based on self-similarity of vision contents. *Image and Vision Computer*. Vol. 22.5. 519-527.

Shi, Chunyi, Wang, Jiaqin. (2000). *Mathematical Logic and Set Theory*. Press of TsingHua University. 2000.12. (2nd ed.)

Su, kang, Guan, Shiyi, & Liu, Jian et al. (1997). A Practical normalized cross-correlation scene matching, *Journal of Astronautics*. Vol.18.3. 1-7.

Sun, Bojiao, & Zhou, Donghua. (2008). Rotated image registration method based on NCC [J], *Transducer and Microsystems Technologies*. Vol 27.5.

Ullah F, & Kaneko S. (2004). Using orientation codes for rotation-invariant template matching [J]. *Pattern Recognition*. P201-209. Vol.37.2.

Wang, Jingdong, Xu, Yibin, & Shen, Chunlin. (2005). New scene matching method for arbitrary rotation. *Journal of Nanjing University of Aeronautics and Astronautics*. Vol. 37.1. 6-10.

Wang, Qi, Li, Yanjun. (2006). Research on Inverse Log- polar Transformation Based on Sub-pixel Interpolation. *Computer Engineering and Application*. Vol 27. 6-7.

Zhao, Fengwei, Li, Jicheng, & Shen, Zhenkang. (2002). Study of scene matching techniques. *Systems Engineering and Electronics*. Vol. 24.12. 110-113.

Table 1. Summary of the experiment 1

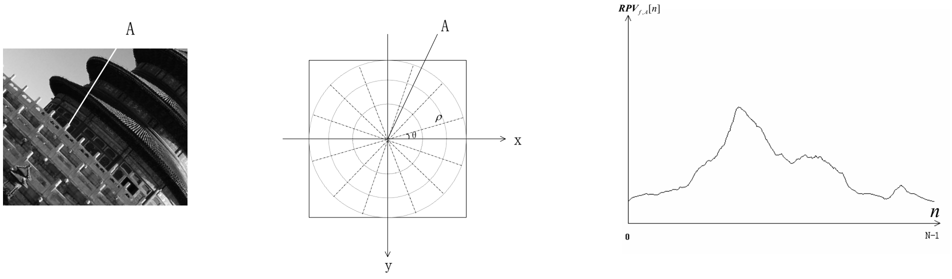
Input	Reference image		(a)	Edge of(a)	(c)	Edge of(c)	(e)	Edge of(e)	(g)	Edge of(g)
	Template image		(b)	Edge of(b)	(d)	Edge of(d)	(f)	Edge of(f)	(h)	Edge of(h)
	Image size	Reference image	600×500		428×321		428×321		250×250	
		Template image	300×250		350×250		269×250		213×212	
Results	Points in SSMP	Reference image	1090	1633	347	303	2455	1234	253	293
		Template image	34	8	27	16	123	81	30	24
	Matching accuracy		97.6%		99.3%		98.1%		99.5%	
	Average Time	Extract SSMP from template images	125 ms		156 ms		141 ms		47 ms	
		Extract template edge images	31 ms		37 ms		31 ms		26 ms	
		Extract SSMP from template-edge images	141 ms		188 ms		171 ms		60 ms	
		RPFT in template and template-edge images	62 ms		63 ms		63 ms		62ms	
		Matching	204 ms		62 ms		328 ms		47ms	
		Determine Rotate angle	162ms		167 ms		171ms		156ms	
		Total time	725 ms		673ms		905ms		398ms	

Table 2. Summary of the experiment 2

Input	Reference image		(a)	Edge of(a)	(c)	Edge of(c)	(e)	Edge of(e)	(g)	Edge of(g)
	Template image		(b)	Edge of(b)	(d)	Edge of(d)	(f)	Edge of(f)	(h)	Edge of(h)
	Image size	Reference image	600×500		428×321		428×321		250×250	
		Template image	300×250		350×250		269×250		213×212	
Results	Points in SSMP	Reference image	1090	1633	347	303	2455	1234	253	293
		Template image	35	8	31	19	128	83	33	25
	Matching accuracy		87.8%		85.3%		86.5%		92.7%	
	Total match time		723 ms		673 ms		897 ms		399ms	

Table 3. Summary of the experiment 3

Input	Reference image		(a)	Edge of(a)	(c)	Edge of(c)	(e)	Edge of(e)	(g)	Edge of(g)
	Template image		(b)	Edge of(b)	(d)	Edge of(d)	(f)	Edge of(f)	(h)	Edge of(h)
	Image size	Reference image	600×500		428×321		428×321		250×250	
		Template image	300×250		350×250		269×250		213×212	
Results	Points in SSMP	Reference image	1090	1633	347	303	2455	1234	253	293
		Template image	50	14	56	32	155	104	31	23
	Matching accuracy		91.6%		92.2%		89.2%		94.1%	
	Total match time		725 ms		680 ms		901 ms		399ms	



(a) grey-level image (b) Radial Projection (c) $\boldsymbol{RPV}_{f,A}$

Figure 1. Schematic diagram of RPV

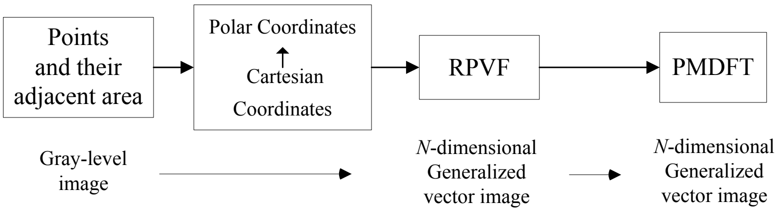

Points and their adjacent area
Polar Coordinates
Cartesian Coordinates
RPVF
PMDFT
Gray-level image
N-dimensional Generalized vector image
N-dimensional Generalized vector image

Figure 2. Overview of RPFT

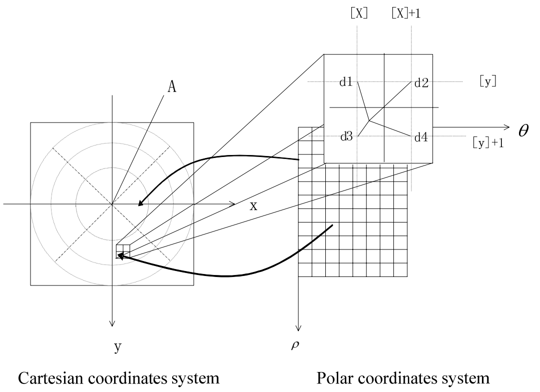
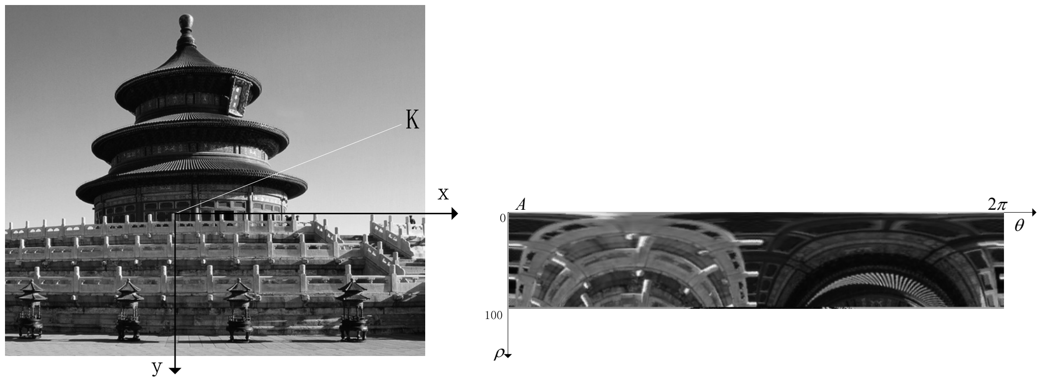
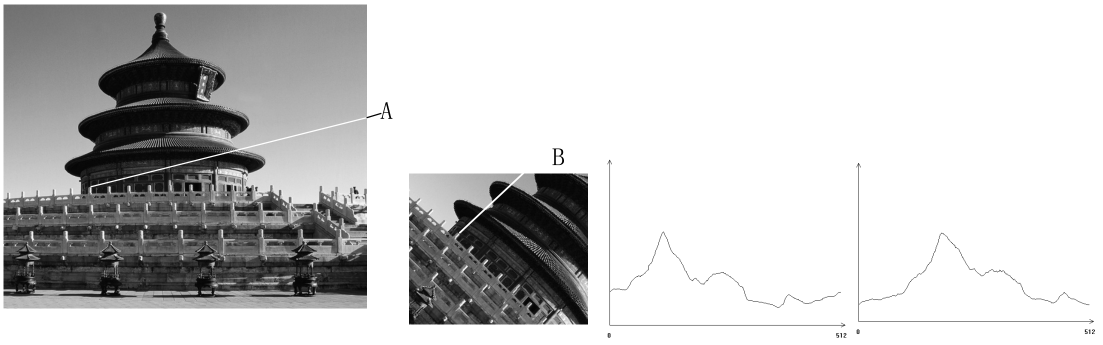


Figure 3. Schematic diagram of inverse transform in polar coordinates



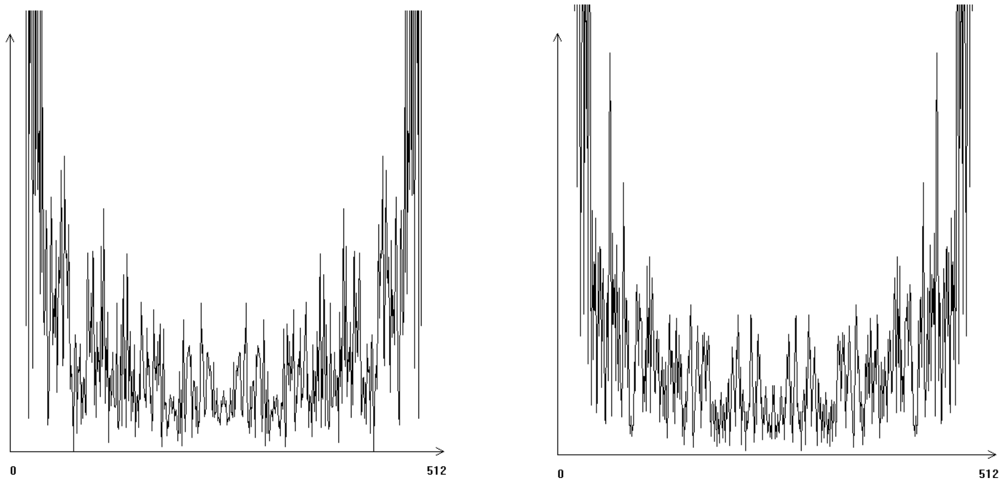
(a) (b)

Figure 4. Inverse transform in polar coordinates



(a) (b) (c) (d)

Figure 5. transform rotation angle to circular shift of RPV



(a) (b)

Figure 6. Rotation invariance of MDFT on RPVs

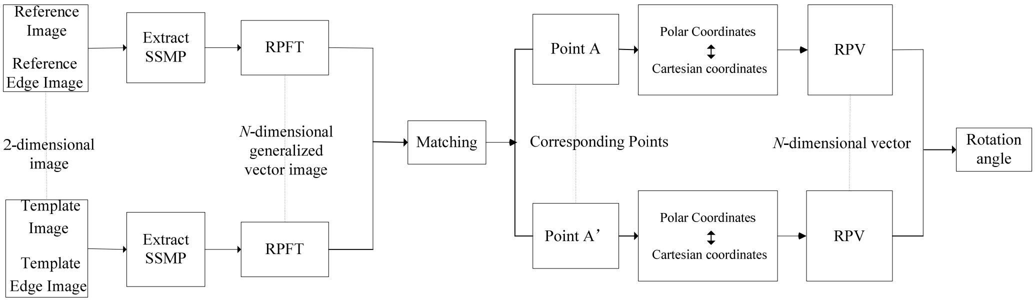


Figure 7. Overview of the ASMRI based on RPFT

1	1	1
1	-2	1
-1	-1	-1

1	1	1
1	-2	-1
1	-1	-1

1	1	-1
1	-2	-1
1	1	-1

1	-1	-1
1	-2	-1
1	1	1

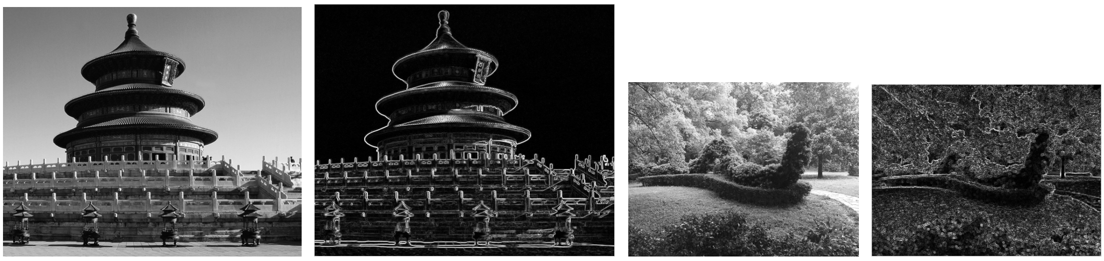
-1	-1	-1
1	-2	1
1	1	1

-1	-1	1
-1	-2	1
1	1	1

-1	1	1
-1	-2	1
-1	1	1

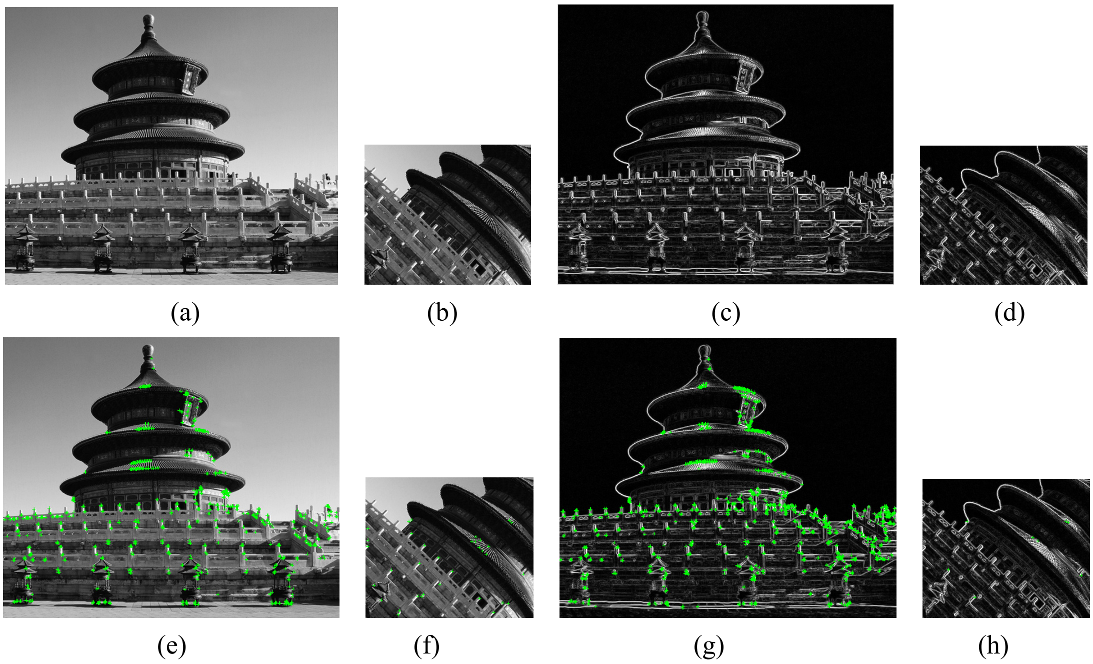
1	1	1
-1	-2	1
-1	-1	1

Figure 8. Templates of Prewitt with orientation



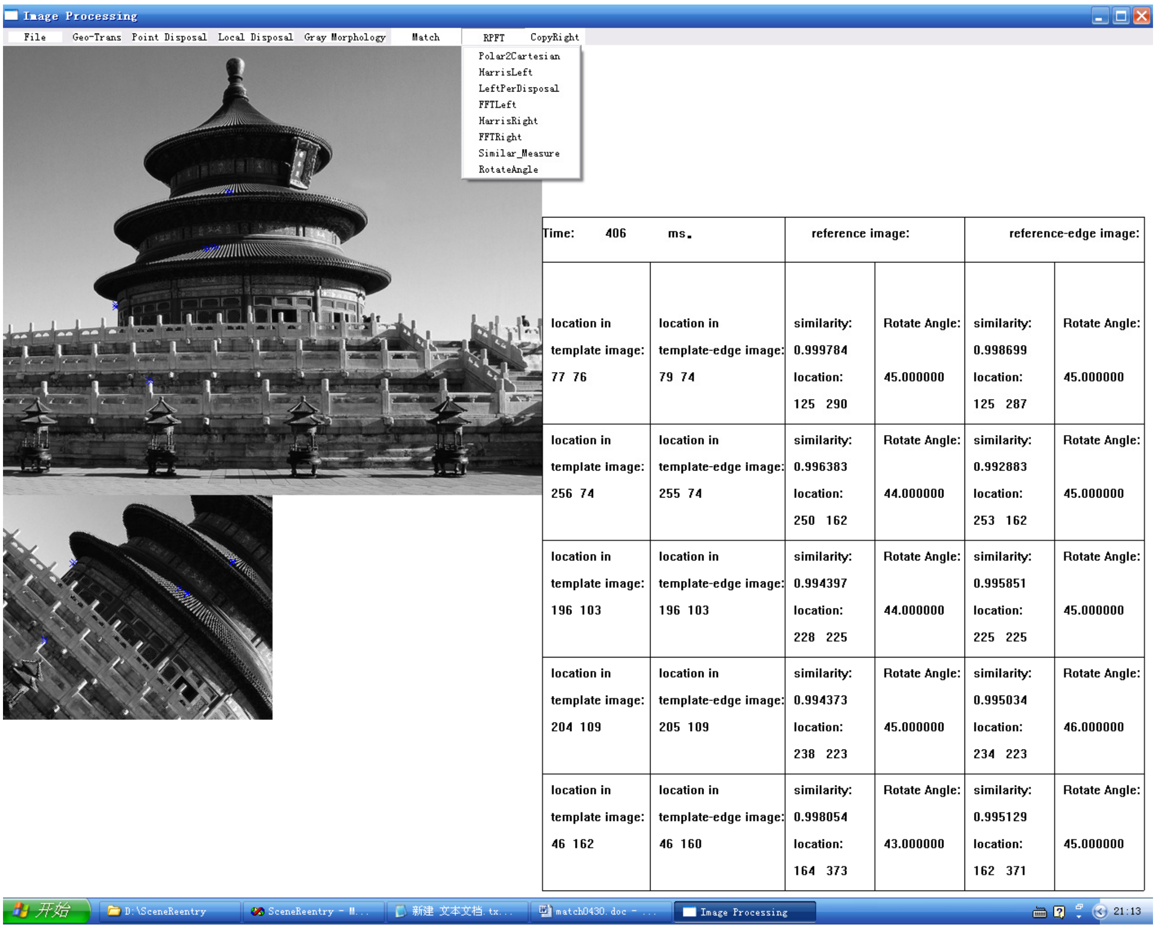
(a) (b) (c) (d)

Figure 9. Extraction of edge images using the operator of Prewitt with orientation


(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 10. Schematic diagram of steps of extracting SSMP



Time: 406 ms.		reference image:		reference-edge image:	
location in template image: 77 76	location in template-edge image: 79 74	similarity: 0.999784 location: 125 290	Rotate Angle: 45.000000	similarity: 0.998699 location: 125 287	Rotate Angle: 45.000000
location in template image: 256 74	location in template-edge image: 255 74	similarity: 0.996383 location: 250 162	Rotate Angle: 44.000000	similarity: 0.992883 location: 253 162	Rotate Angle: 45.000000
location in template image: 196 103	location in template-edge image: 196 103	similarity: 0.994397 location: 228 225	Rotate Angle: 44.000000	similarity: 0.995851 location: 225 225	Rotate Angle: 45.000000
location in template image: 204 109	location in template-edge image: 205 109	similarity: 0.994373 location: 238 223	Rotate Angle: 45.000000	similarity: 0.995034 location: 234 223	Rotate Angle: 46.000000
location in template image: 46 162	location in template-edge image: 46 160	similarity: 0.998054 location: 164 373	Rotate Angle: 43.000000	similarity: 0.995129 location: 162 371	Rotate Angle: 45.000000

Figure 11. The interface of the program

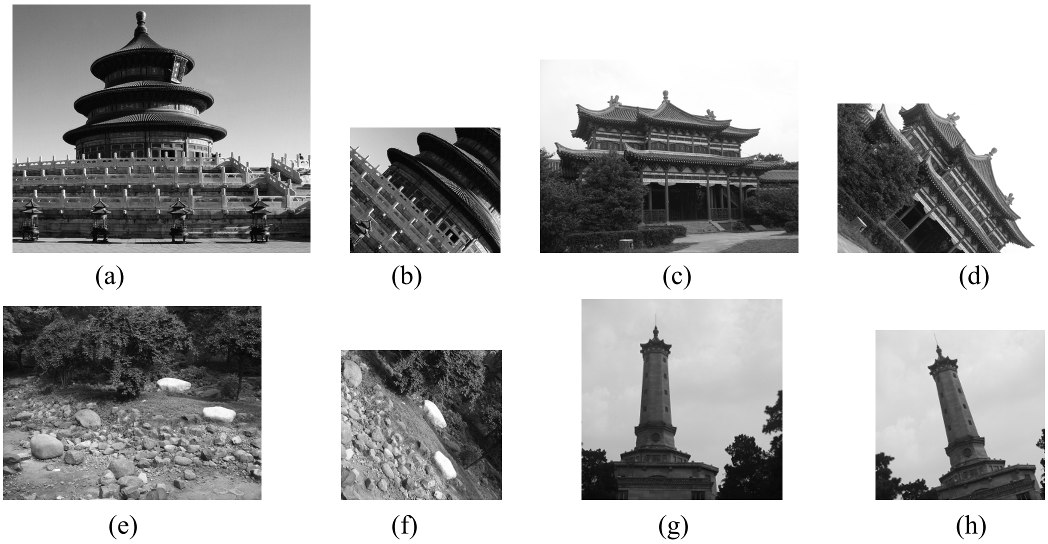


Figure 12. rotation between reference and template images

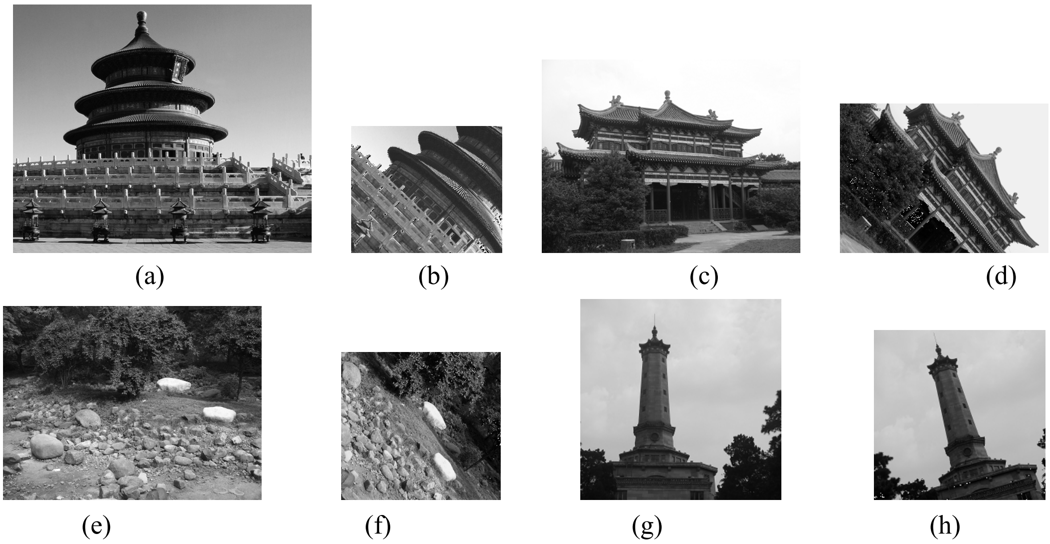


Figure 13. rotation and grey change between reference and template images

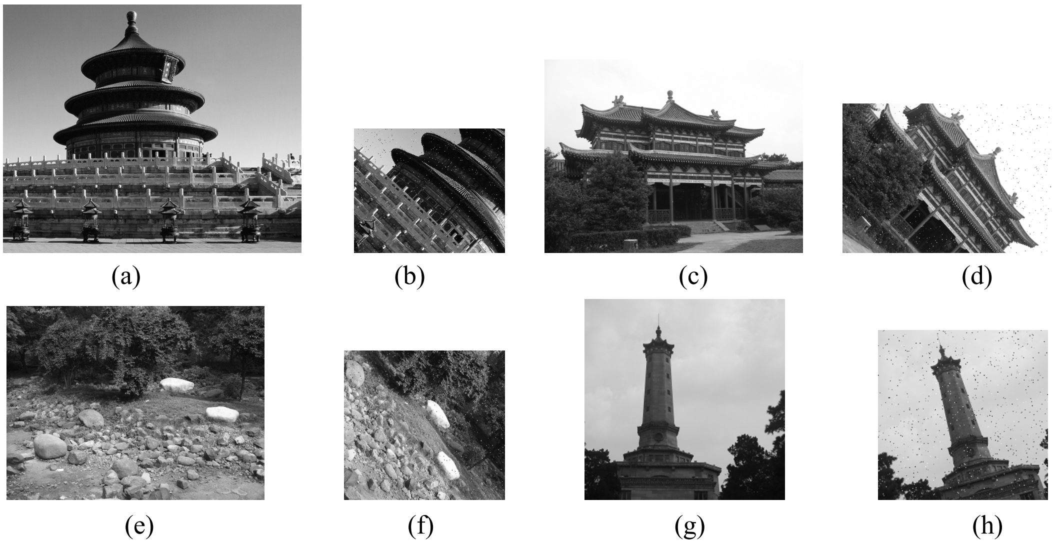


Figure 14. rotation and Gaussian noise between reference and template images