Thus we may solve for the 6 unknowns by *Cramer's rule* (see p.89 of [8]) as follows:

$$\begin{aligned} a_{11} &= \det(M_1)/\det(M), & a_{12} &= \det(M_2)/\det(M), \\ a_{21} &= \det(M_3)/\det(M), & a_{22} &= \det(M_4)/\det(M), \\ b_1 &= \det(M_5)/\det(M), & b_2 &= \det(M_6)/\det(M), \end{aligned}$$

where $M = [m_{ij}]_{i,j=1,\dots,6}$ and $M_k = M$ with k-th column substituted by $[n_i]^t_{i=1,\dots,6}$, for $k = 1, \dots, 6$.

Discussion
We might wish to do a *weighted* sum of the squared distances. More specifically, the 2 squared distances provided by the endpoints of a long line segments will get more weight than those of short line segments.

6 Acknowledgement

The author would like to thank Professor Jack Schwartz for generous guidance and Professor Jiawei Hong for discussions on best least-squares match procedures. Many thanks go to his colleague Mr. Ronie Hecker for bringing parameter space recognition to his attention.

References

[1] Duda, R. O. and P. E. Hart, 1972. *Use of the Hough Transform to Detect Lines and Curves in Pictures*, Communication of ACM, vol. 15, pp. 11–15.

[2] Grimson, W. E.L. and D. P. Huttenlocher, 1990. *On the Sensitivity of Geometric Hashing*, Proc. of the 3rd Int. Conf. on Computer Vision, pp. 334–338.

[3] Hecker, Y. C. and R. M. Bolle, 1991. *Invariant Feature Matching in Parameter Space with Application to Line Features*, Technical Report RC 16447 IBM T.J.Watson Research Center.

[4] Klein, F., 1925. *Elementary Mathematics from an Advanced Standpoint: Geometry*, Macmillan, New York, 1925 (Third Edition). (English translation).

[5] Lamdan, Y., J. T. Schwartz and H. J. Wolfson, 1988. *On Recognition of 3-D Objects from 2-D Images*, Proc. of the IEEE Int. Conf. on Robotics and Automation, pp. 1407–1413.

[6] Lamdan, Y., J. T. Schwartz and H. J. Wolfson, 1988. *Object Recognition by Affine Invariant Matching*, Proc. of the IEEE Int. Conf. on Computer Vision and Pattern, pp. 335–344.

[7] Lamdan, Y. 1989. *Object Recognition by Geometric Hashing*, Ph.D. Thesis, Computer Science Division, Courant Institute of Mathematical Sciences, New York University.

[8] Lang, S. *Linear Algebra*, Addison-Wesley, 1966.

and

$$E = min_{\mathbf{T}} \sum_{j=1}^{n} ((\cos\theta_j, \sin\theta_j)\mathbf{T}(\mathbf{u}_{j1}) - r_j)^2 + ((\cos\theta_j, \sin\theta_j)\mathbf{T}(\mathbf{u}_{j2}) - r_j)^2$$

To minimize E, we have to solve the following system of equations:

$$\frac{\partial E}{\partial a_{11}} = 0, \quad \frac{\partial E}{\partial a_{12}} = 0, \quad \frac{\partial E}{\partial a_{21}} = 0, \quad \frac{\partial E}{\partial a_{22}} = 0, \quad \frac{\partial E}{\partial b_1} = 0 \quad \text{and} \quad \frac{\partial E}{\partial b_2} = 0.$$

Since E is a quadratic function in each of its unknowns, the above is a system of linear equations with six unknowns as follows in matrix form:

$$\begin{pmatrix} m_{11} & m_{12} & m_{13} & m_{14} & m_{15} & m_{16} \\ m_{21} & m_{22} & m_{23} & m_{24} & m_{25} & m_{26} \\ m_{31} & m_{32} & m_{33} & m_{34} & m_{35} & m_{36} \\ m_{41} & m_{42} & m_{43} & m_{44} & m_{45} & m_{46} \\ m_{51} & m_{52} & m_{53} & m_{54} & m_{55} & m_{56} \\ m_{61} & m_{62} & m_{63} & m_{64} & m_{65} & m_{66} \end{pmatrix} \begin{pmatrix} a_{11} \\ a_{12} \\ a_{21} \\ a_{22} \\ b_1 \\ b_2 \end{pmatrix} = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \\ n_4 \\ n_5 \\ n_6 \end{pmatrix}$$

where

$m_{11} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (x_{j1}^2 + x_{j2}^2)$,
$m_{12} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{13} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}^2 + x_{j2}^2)$,
$m_{14} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{15} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (x_{j1} + x_{j2})$,
$m_{16} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1} + x_{j2})$,

$m_{31} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}^2 + x_{j2}^2)$,
$m_{32} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{33} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (x_{j1}^2 + x_{j2}^2)$,
$m_{34} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{35} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1} + x_{j2})$,
$m_{36} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (x_{j1} + x_{j2})$,

$m_{21} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{22} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (y_{j1}^2 + y_{j2}^2)$,
$m_{23} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{24} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1}^2 + y_{j2}^2)$,
$m_{25} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (y_{j1} + y_{j2})$,
$m_{26} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1} + y_{j2})$,

$m_{41} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{42} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1}^2 + y_{j2}^2)$,
$m_{43} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (x_{j1}y_{j1} + x_{j2}y_{j2})$,
$m_{44} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (y_{j1}^2 + y_{j2}^2)$,
$m_{45} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1} + y_{j2})$,
$m_{46} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (y_{j1} + y_{j2})$,

$m_{51} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (x_{j1} + x_{j2})$,
$m_{52} = 2\sum_{j=1}^{n} \cos^2\theta_j \, (y_{j1} + y_{j2})$,
$m_{53} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1} + x_{j2})$,
$m_{54} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1} + y_{j2})$,
$m_{55} = 4\sum_{j=1}^{n} \cos^2\theta_j$,
$m_{56} = 4\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j$,

$m_{61} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (x_{j1} + x_{j2})$,
$m_{62} = 2\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j \, (y_{j1} + y_{j2})$,
$m_{63} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (x_{j1} + x_{j2})$,
$m_{64} = 2\sum_{j=1}^{n} \sin^2\theta_j \, (y_{j1} + y_{j2})$,
$m_{65} = 4\sum_{j=1}^{n} \cos\theta_j \, \sin\theta_j$,
$m_{66} = 4\sum_{j=1}^{n} \sin^2\theta_j$

$n_1 = 2\sum_{j=1}^{n} \cos\theta_j \, r_j(x_{j1} + x_{j2})$,
$n_2 = 2\sum_{j=1}^{n} \cos\theta_j \, r_j(y_{j1} + y_{j2})$,
$n_5 = 4\sum_{j=1}^{n} \cos\theta_j \, r_j$,

$n_3 = 2\sum_{j=1}^{n} \sin\theta_j \, r_j(x_{j1} + x_{j2})$,
$n_4 = 2\sum_{j=1}^{n} \sin\theta_j \, r_j(y_{j1} + y_{j2})$,
$n_6 = 4\sum_{j=1}^{n} \sin\theta_j \, r_j$.

In this section, we discuss several heuristics to minimize the errors.

Heuristic 1
Treat each line as a point with coordinate (θ, r) in (θ, r)-space and minimize the squared distance between (θ, r) and its correspondence (θ', r').

Discussion
The problem of this heuristic is that θ and r are of different metrics. To minimize the squared distance between a point (θ, r) and its correspondence (θ', r'), we implicitly assume equal weight on both θ and r.

Heuristic 2
To circumvent the problem caused by **Heuristic 1**, we note that a line (θ, r) can be uniquely represented by the point $(r\cos\theta, r\sin\theta)$, which is the projection of the origin onto the line. To match line (θ, r) to its correspondence (θ', r'), we try to minimize the squared distance between $(r\cos\theta, r\sin\theta)$ and $(r'\cos\theta', r'\sin\theta')$.

Discussion
The drawback of this heuristic is its dependency on the origin. The nearer the line is to the origin, the more weight is on r (think of the special case when both lines pass through the origin in the image).

Heuristic 3[2]
The models in the model base are usually finite in the sense that though they are modeled by lines, they in fact consist of line *segments*. We may work in the image space by minimizing the squared distance of the endpoints of the transformed model line segments to their corresponding scene lines.

We derived in the following the closed-form formula for the case of affine transformations, which include rigid and similarity transformations. That for projective transformations is yet to be attempted.

Specifically, assuming that we are looking for an affine match between n scene lines l_j and endpoints of n segments, $\mathbf{u}_{j1}$ and $\mathbf{u}_{j2}$, $j = 1, \ldots, n$, we would like to find the affine transformation $\mathbf{T} = (\mathbf{A}, \mathbf{b})$, such that the summations of the squared distances between the sequence $\mathbf{T}(\mathbf{u}_{j1})$ to l_j and $\mathbf{T}(\mathbf{u}_{j2})$ to l_j, $j = 1, \ldots, n$, is minimized:

$$E = min_{\mathbf{T}} \sum_{j=1}^{n} (\text{distance of } \mathbf{T}(\mathbf{u}_{j1}) \text{ and } l_j)^2 + (\text{distance of } \mathbf{T}(\mathbf{u}_{j2}) \text{ and } l_j)^2.$$

Let line l_j be with parameter (θ_j, r_j) and endpoints $\mathbf{u}_{ji}$ be (x_{ji}, y_{ji}), $j = 1, \ldots, n$ and $i = 1, 2$. Also let $\mathbf{T} = (\mathbf{A}, \mathbf{b})$ such that

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{pmatrix} \quad \text{and} \quad \mathbf{b} = \begin{pmatrix} b_1 \\ b_2 \end{pmatrix}.$$

Then

$$\mathbf{T}(\mathbf{u}_{ji}) = (a_{11}x_{ji} + a_{12}y_{ji} + b_1, a_{21}x_{ji} + a_{22}y_{ji} + b_2)^t$$

[2] This is suggested by Professor Jiawei Hong, affiliated with CIMS, NYU, New York.

$$
\begin{aligned}
p_{22} &= -C_1C_3(r_1\cos\theta_3 - r_3\cos\theta_1),\\
p_{23} &= -C_1C_3\sin(\theta_1-\theta_3),\\
p_{31} &= C_1C_2(r_1\sin\theta_2 - r_2\sin\theta_1),\\
p_{32} &= -C_1C_2(r_1\cos\theta_2 - r_2\cos\theta_1),\\
p_{33} &= -C_1C_2\sin(\theta_1-\theta_2).
\end{aligned}
$$

From section 3.3, the invariant $(\theta', r')^t$ of a line $(\theta, r)^t$ with respect to the basis lines can be obtained as follows:

$$(\theta', r')^t = \mathbf{H}((\theta, r)^t) = \mathbf{G}(\mathbf{P}(\cos\theta, \sin\theta, -r)^t) = \mathbf{G}(\mathbf{a}')$$

where

$$\mathbf{a}' = -1/C_1C_2C_3\lambda_4 \begin{pmatrix} ABC \\ DEF \\ GCF \end{pmatrix}_{3\times 1}$$

where

$$
\begin{aligned}
A &= r_1\sin(\theta_2-\theta_3) + r_2\sin(\theta_3-\theta_1) + r_3\sin(\theta_1-\theta_2),\\
B &= r_4\sin(\theta-\theta_2) - r_2\sin(\theta-\theta_4) + r\sin(\theta_2-\theta_4),\\
C &= r_4\sin(\theta_1-\theta_3) - r_3\sin(\theta_1-\theta_4) + r_1\sin(\theta_3-\theta_4),\\
D &= -r_3\sin(\theta-\theta_1) + r_1\sin(\theta-\theta_3) - r\sin(\theta_1-\theta_3),\\
E &= r_4\sin(\theta_1-\theta_2) - r_2\sin(\theta_1-\theta_4) + r_1\sin(\theta_2-\theta_4),\\
F &= r_4\sin(\theta_2-\theta_3) - r_3\sin(\theta_2-\theta_4) + r_2\sin(\theta_3-\theta_4),\\
G &= r_2\sin(\theta-\theta_1) - r_1\sin(\theta-\theta_2) + r\sin(\theta_1-\theta_2),
\end{aligned}
$$

and hence

$$
\begin{aligned}
\theta' &= \tan^{-1}(DEF/ABC),\\
r' &= \frac{-GCF}{\sqrt{(ABC)^2 + (DEF)^2}}.
\end{aligned}
$$

5 Best Least-Squares Match

The transformation between a model and its scene instance can be recovered by the correspondence of the model basis and the scene basis *alone*. However, scene lines detected by the *Hough* transform may be somewhat distorted due to noise, which results in distortion of the computation of the transformation. Usually this distorted transformation transforms a model to match its scene instance with basis lines matching each other perfectly while the other lines deviating from their correspondences more or less. Knowledge of additional line correspondences between a model and its scene instance can be used to improve the accuracy of the computed transformation.

Since $ax + by + c = 0$ and $\lambda ax + \lambda by + \lambda c = 0$, $\lambda \neq 0$, represent the same line, we have

$$\begin{aligned} \lambda_1 \mathbf{P} \mathbf{a}_1 &= \mathbf{e}_1, \\ \lambda_2 \mathbf{P} \mathbf{a}_2 &= \mathbf{e}_2, \\ \lambda_3 \mathbf{P} \mathbf{a}_3 &= \mathbf{e}_3, \\ \lambda_4 \mathbf{P} \mathbf{a}_4 &= \mathbf{e}_4, \end{aligned}$$

where λ_1, λ_2, λ_3 and λ_4 are non-zero constants. Equivalently,

$$\mathbf{P}(\lambda_1 \mathbf{a}_1, \lambda_2 \mathbf{a}_2, \lambda_3 \mathbf{a}_3, \lambda_4 \mathbf{a}_4) = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_4).$$

Then

$$\begin{aligned} (\lambda_1 \mathbf{a}_1, \lambda_2 \mathbf{a}_2, \lambda_3 \mathbf{a}_3, \lambda_4 \mathbf{a}_4) &= \mathbf{P}^{-1}(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3, \mathbf{e}_4) \\ &= \begin{pmatrix} a_{11} & a_{21} & a_{11} - a_{31} & a_{21} - a_{31} \\ a_{12} & a_{22} & a_{12} - a_{32} & a_{22} - a_{32} \\ a_{13} & a_{23} & a_{13} - a_{33} & a_{23} - a_{33} \end{pmatrix}. \end{aligned}$$

We have

$$\lambda_1 = \frac{a_{11}}{\cos\theta_1} = \frac{a_{12}}{\sin\theta_1} = \frac{a_{13}}{-r_1} \tag{16}$$

$$\lambda_2 = \frac{a_{21}}{\cos\theta_2} = \frac{a_{22}}{\sin\theta_2} = \frac{a_{23}}{-r_2} \tag{17}$$

$$\lambda_3 = \frac{a_{11} - a_{31}}{\cos\theta_3} = \frac{a_{12} - a_{32}}{\sin\theta_3} = \frac{a_{13} - a_{33}}{-r_3} \tag{18}$$

$$\lambda_4 = \frac{a_{21} - a_{31}}{\cos\theta_4} = \frac{a_{22} - a_{32}}{\sin\theta_4} = \frac{a_{23} - a_{33}}{-r_4} \tag{19}$$

From (16), (17), (18) and (19), we may solve a_{ij}, $i, j = 1, 2, 3$, in terms of λ_4. Substituting them in (15), we obtain

$$\mathbf{P} = -1/C_1 C_2 C_3 \lambda_4 \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix}$$

where

$$\begin{aligned} C_1 &= -r_4 \sin(\theta_2 - \theta_3) + r_3 \sin(\theta_2 - \theta_4) - r_2 \sin(\theta_3 - \theta_4), \\ C_2 &= -r_4 \sin(\theta_1 - \theta_3) + r_3 \sin(\theta_1 - \theta_4) - r_1 \sin(\theta_3 - \theta_4), \\ C_3 &= r_4 \sin(\theta_1 - \theta_2) - r_2 \sin(\theta_1 - \theta_4) + r_1 \sin(\theta_2 - \theta_4), \\ p_{11} &= C_2(C_1 r_2 \sin\theta_1 - C_1 r_1 \sin\theta_2 + C_3 r_3 \sin\theta_2 - C_3 r_2 \sin\theta_3), \\ p_{12} &= -C_2(C_1 r_2 \cos\theta_1 - C_1 r_1 \cos\theta_2 + C_3 r_3 \cos\theta_2 - C_3 r_2 \cos\theta_3), \\ p_{13} &= C_2(C_1 \sin(\theta_1 - \theta_2) + C_3 \sin(\theta_2 - \theta_3)), \\ p_{21} &= C_1 C_3 (r_1 \sin\theta_3 - r_3 \sin\theta_1), \end{aligned}$$

To determine a projective transformation uniquely, we need a correspondence of two ordered quadruplets of lines and for each quadruplet, no three lines are parallel to one another or intersect at a common point (thus no three points are collinear in *parameter space*).

One way to encode the fifth line in a projective-invariant way is to find a transformation $\mathbf{P}$, which maps the four basis lines to a canonical basis, say $x = 0$, $y = 0$, $x = 1$ and $y = 1$, then apply $\mathbf{P}$ to the fifth line.

A projective transformation $\mathbf{T}$ in image space is of the form

$$\mathbf{T} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}.$$

This corresponds to the parameter-space transformation $\mathbf{P}$ defined by,

$$\mathbf{P} = (\mathbf{T}^{-1})^t = \frac{1}{D} \begin{pmatrix} a_{22}a_{33} - a_{23}a_{32} & a_{23}a_{31} - a_{21}a_{33} & a_{21}a_{32} - a_{22}a_{31} \\ a_{13}a_{32} - a_{12}a_{33} & a_{11}a_{33} - a_{13}a_{31} & a_{12}a_{31} - a_{11}a_{32} \\ a_{12}a_{23} - a_{13}a_{22} & a_{13}a_{21} - a_{11}a_{23} & a_{11}a_{22} - a_{12}a_{21} \end{pmatrix}, \tag{15}$$

where

$$D = a_{11}a_{22}a_{33} + a_{12}a_{23}a_{31} + a_{13}a_{21}a_{32} - a_{11}a_{23}a_{32} - a_{12}a_{21}a_{33} - a_{13}a_{22}a_{31}$$

so that

$$\mathbf{P}^{-1} = \mathbf{T}^t = \begin{pmatrix} a_{11} & a_{21} & a_{31} \\ a_{12} & a_{22} & a_{32} \\ a_{13} & a_{23} & a_{33} \end{pmatrix}.$$

Let a 2-D line be parameterized as $\cos\theta\, x + \sin\theta\, y - r = 0$ and let four additional basis lines be represented by their parameters $\mathbf{a}_1$, $\mathbf{a}_2$, $\mathbf{a}_3$ and $\mathbf{a}_4$, s.t.

$$\begin{aligned} \mathbf{a}_1 &= (\cos\theta_1, \sin\theta_1, -r_1)^t, \\ \mathbf{a}_2 &= (\cos\theta_2, \sin\theta_2, -r_2)^t, \\ \mathbf{a}_3 &= (\cos\theta_3, \sin\theta_3, -r_3)^t, \\ \mathbf{a}_4 &= (\cos\theta_4, \sin\theta_4, -r_4)^t, \end{aligned}$$

which are to be mapped by $\mathbf{P}$ to

$$\begin{aligned} \mathbf{e}_1 &= (\cos 0, \sin 0, 0)^t = (1, 0, 0)^t, \\ \mathbf{e}_2 &= (\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, 0)^t = (0, 1, 0)^t, \\ \mathbf{e}_3 &= (\cos 0, \sin 0, -1)^t = (1, 0, -1)^t, \\ \mathbf{e}_4 &= (\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, -1)^t = (0, 1, -1)^t. \end{aligned}$$

Our intention is to map the first basis line to y-axis; the second basis line to x-axis; the third basis line to $x = 1$; the fourth basis line to $y = 1$.

Thus

$$\lambda_1 = \frac{a_{11}}{\cos\theta_1} = \frac{a_{12}}{\sin\theta_1} = \frac{b_1}{-r_1}, \tag{10}$$

$$\lambda_2 = \frac{a_{21}}{\cos\theta_2} = \frac{a_{22}}{\sin\theta_2} = \frac{b_2}{-r_2}, \tag{11}$$

$$\lambda_3 = \frac{\frac{1}{\sqrt{2}}(a_{11}+a_{21})}{\cos\theta_3} = \frac{\frac{1}{\sqrt{2}}(a_{12}+a_{22})}{\sin\theta_3} = \frac{\frac{1}{\sqrt{2}}(b_1+b_2-1)}{-r_3}. \tag{12}$$

From (10), (11) and (12), we may solve a_1, a_2, a_3, a_4, b_1 and b_2. Substituting them in (9), we obtain

$$\mathbf{P} = \begin{pmatrix} \csc(\theta_1-\theta_2)\csc(\theta_2-\theta_3)\sin\theta_2 A & -\csc(\theta_1-\theta_2)\csc(\theta_2-\theta_3)\cos\theta_2 A & 0 \\ \csc(\theta_1-\theta_2)\csc(\theta_1-\theta_3)\sin\theta_1 A & -\csc(\theta_1-\theta_2)\csc(\theta_1-\theta_3)\cos\theta_1 A & 0 \\ \csc(\theta_1-\theta_2)(r_2\sin\theta_1 - r_1\sin\theta_2) & -\csc(\theta_1-\theta_2)(r_2\cos\theta_1 - r_1\cos\theta_2) & 1 \end{pmatrix}$$

where

$$A = r_1\sin(\theta_2-\theta_3) + r_2\sin(\theta_3-\theta_1) + r_3\sin(\theta_1-\theta_2).$$

From section 3.3, the invariant $(\theta', r')^t$ of a line $(\theta, r)^t$ with respect to the basis lines can be obtained as follows:

$$(\theta', r')^t = \mathbf{H}((\theta, r)^t) = \mathbf{G}(\mathbf{P}(\cos\theta, \sin\theta, -r)^t) = \mathbf{G}(\mathbf{a}')$$

where

$$\mathbf{a}' = \begin{pmatrix} -\csc(\theta_1-\theta_2)\csc(\theta_2-\theta_3)\sin(\theta-\theta_2)A \\ \\ -\csc(\theta_1-\theta_2)\csc(\theta_1-\theta_3)\sin(\theta-\theta_1)A \\ \\ r_1\csc(\theta_1-\theta_2)\sin(\theta-\theta_2) - r_2\csc(\theta_1-\theta_2)\sin(\theta-\theta_1) - r \end{pmatrix}_{3\times 1}$$

and hence

$$\theta' = \tan^{-1}\left(\frac{\csc(\theta_1-\theta_3)\sin(\theta_1-\theta)}{\csc(\theta_2-\theta_3)\sin(\theta_2-\theta)}\right), \tag{13}$$

$$r' = \frac{1}{\tau}(r_1\csc(\theta_1-\theta_2)\sin(\theta_2-\theta) - r_2\csc(\theta_1-\theta_2)\sin(\theta_1-\theta) + r) \tag{14}$$

where

$$\tau = \sqrt{\csc^2(\theta_1-\theta_3)\sin^2(\theta_1-\theta) + \csc^2(\theta_2-\theta_3)\sin^2(\theta_2-\theta)}\,|\csc(\theta_1-\theta_2)A|.$$

4.5 Line Invariants under Projective Transformations

Although a projective transformation is fractional linear, one can well treat it as a linear transformation by using homogeneous coordinates.

An affine transformation $\mathbf{T}$ in image space is of the form

$$\mathbf{T} = \begin{pmatrix} \mathbf{A} & \mathbf{b} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & b_1 \\ a_{21} & a_{22} & b_2 \\ 0 & 0 & 1 \end{pmatrix}$$

where $\mathbf{A}$ is the skewing matrix; $\mathbf{b}$, the translation vector. This corresponds to the parameter-space transformation $\mathbf{P}$ defined by,

$$\mathbf{P} = (\mathbf{T}^{-1})^t = \begin{pmatrix} (\mathbf{A}^{-1})^t & 0 \\ -\mathbf{b}^t(\mathbf{A}^{-1})^t & 1 \end{pmatrix}, \tag{9}$$

so that

$$\mathbf{P}^{-1} = \mathbf{T}^t = \begin{pmatrix} \mathbf{A}^t & 0 \\ \mathbf{b}^t & 1 \end{pmatrix}.$$

Let a 2-D line be parameterized as $\cos\theta\, x + \sin\theta\, y - r = 0$ and let three additional basis lines be represented by their parameters $\mathbf{a}_1$, $\mathbf{a}_2$ and $\mathbf{a}_3$, s.t.

$$\begin{aligned} \mathbf{a}_1 &= (\cos\theta_1, \sin\theta_1, -r_1)^t, \\ \mathbf{a}_2 &= (\cos\theta_2, \sin\theta_2, -r_2)^t, \\ \mathbf{a}_3 &= (\cos\theta_3, \sin\theta_3, -r_3)^t, \end{aligned}$$

which are to be mapped by $\mathbf{P}$ to

$$\begin{aligned} \mathbf{e}_1 &= (\cos 0, \sin 0, 0)^t = (1, 0, 0)^t, \\ \mathbf{e}_2 &= (\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, 0)^t = (0, 1, 0)^t, \\ \mathbf{e}_3 &= (\cos\frac{\pi}{4}, \sin\frac{\pi}{4}, \frac{-1}{\sqrt{2}})^t = (\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, \frac{-1}{\sqrt{2}})^t. \end{aligned}$$

Our intention is to map the first basis line to y-axis; the second basis line to x-axis; the third basis line to $x + y = 1$.

Since $ax + by + c = 0$ and $\lambda ax + \lambda by + \lambda c = 0$, $\lambda \neq 0$, represent the same line, we have

$$\begin{aligned} \lambda_1 \mathbf{P} \mathbf{a}_1 &= \mathbf{e}_1, \\ \lambda_2 \mathbf{P} \mathbf{a}_2 &= \mathbf{e}_2, \\ \lambda_3 \mathbf{P} \mathbf{a}_3 &= \mathbf{e}_3, \end{aligned}$$

where λ_1, λ_2 and λ_3 are non-zero constants. Equivalently,

$$\mathbf{P}(\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2, \lambda_3\mathbf{a}_3) = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3).$$

Then

$$\begin{aligned} (\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2, \lambda_3\mathbf{a}_3) &= \mathbf{P}^{-1}(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3) \\ &= \begin{pmatrix} a_{11} & a_{21} & \frac{1}{\sqrt{2}}(a_{11} + a_{21}) \\ a_{12} & a_{22} & \frac{1}{\sqrt{2}}(a_{12} + a_{22}) \\ b_1 & b_2 & \frac{1}{\sqrt{2}}(b_1 + b_2 - 1) \end{pmatrix}. \end{aligned}$$

Thus

$$\begin{aligned}
\lambda_1 &= \frac{-s\,\sin\phi}{\cos\theta_1} = \frac{s\,\cos\phi}{\sin\theta_1} = \frac{b_2}{-r_1}, && (6)\\
\lambda_2 &= \frac{s(\cos\phi C_1 - \sin\phi S_1)}{\cos\theta_2} = \frac{s(\sin\phi C_1 + \cos\phi S_1)}{\sin\theta_2} = \frac{b_1 C_1 + b_2 S_1}{-r_2}, && (7)\\
\lambda_3 &= \frac{s(\cos\phi C_2 - \sin\phi S_2)}{\cos\theta_3} = \frac{s(\sin\phi C_2 + \cos\phi S_2)}{\sin\theta_3} = \frac{b_1 C_2 + b_2 S_2 - \sin|\theta_1 - \theta_3|}{-r_3}. && (8)
\end{aligned}$$

From (6), (7) and (8), we may solve ϕ, b_1, b_2 and s. Substituting them in (5), we obtain

$$\mathbf{P} = \frac{1}{D}\begin{pmatrix} \sin\theta_1 A & -\cos\theta_1 A & 0 \\ \cos\theta_1 A & \sin\theta_1 A & 0 \\ (r_2\sin\theta_1 - r_1\sin\theta_2)\sin(\theta_1-\theta_3) & -(r_2\cos\theta_1 - r_1\cos\theta_2)\sin(\theta_1-\theta_3) & D \end{pmatrix}$$

where

$$D = \sin(\theta_1 - \theta_2)\sin(\theta_1 - \theta_3),$$
$$A = r_1\sin(\theta_2 - \theta_3) + r_2\sin(\theta_3 - \theta_1) + r_3\sin(\theta_1 - \theta_2).$$

From section 3.3, the invariant $(\theta', r')^t$ of a line $(\theta, r)^t$ with respect to the basis lines can be obtained as follows:

$$(\theta', r')^t = \mathbf{H}((\theta, r)^t) = \mathbf{G}(\mathbf{P}(\cos\theta, \sin\theta, -r)^t) = \mathbf{G}(\mathbf{a}')$$

where

$$\mathbf{a}' = \frac{1}{D}(-\sin(\theta-\theta_1)A, \cos(\theta-\theta_1)A, (-r_2\sin(\theta-\theta_1) + r_1\sin(\theta-\theta_2) - r\sin(\theta_1-\theta_2))\sin(\theta_1-\theta_3))^t$$

and hence

$$\begin{aligned}
\theta' &= \theta - \theta_1 + \frac{\pi}{2},\\
r' &= (r_2\sin(\theta-\theta_1) - r_1\sin(\theta-\theta_2) + r\sin(\theta_1-\theta_2))\sin(\theta_1-\theta_3)/|A|.
\end{aligned}$$

4.4 Line Invariants under Affine Transformations

Affine transformations are often appropriate approximations to perspective transformations (see p. 79 of [4]) and thus can be used in recognition algorithms as a substitute for more general perspective transformations.

Given $\triangle ABC$ and $\triangle DEF$, there is a *unique* affine transformation $\mathbf{T}$, such that $\mathbf{T}(A) = D$, $\mathbf{T}(B) = E$ and $\mathbf{T}(C) = F$. To determine an affine transformation uniquely, we need a correspondence of two ordered triplets of lines, which are not parallel to one another and do not intersect at a common point.

One way to encode the fourth line in an affine-invariant way is to find a transformation $\mathbf{P}$, which maps the three basis lines to a canonical basis, say $x = 0$, $y = 0$ and $x + y = 1$, then apply $\mathbf{P}$ to the fourth line.

where s is the scaling factor; $\mathbf{R}$, the rotation matrix; $\mathbf{b}$, the translation vector. This corresponds to the parameter-space transformation $\mathbf{P}$ defined by

$$\mathbf{P} = (\mathbf{T}^{-1})^t = \begin{pmatrix} \frac{1}{s}(\mathbf{R}^{-1})^t & 0 \\ -\frac{1}{s}\mathbf{b}^t(\mathbf{R}^{-1})^t & 1 \end{pmatrix} = \begin{pmatrix} \frac{1}{s}\mathbf{R} & 0 \\ -\frac{1}{s}\mathbf{b}^t\mathbf{R} & 1 \end{pmatrix}, \tag{5}$$

so that

$$\mathbf{P}^{-1} = \mathbf{T}^t = \begin{pmatrix} s\mathbf{R}^t & 0 \\ \mathbf{b}^t & 1 \end{pmatrix}.$$

Let a 2-D line be parameterized as $\cos\theta\, x + \sin\theta\, y - r = 0$ and let three additional basis lines, assuming the first line is not parallel to the others, be represented by their parameters $\mathbf{a}_1$, $\mathbf{a}_2$ and $\mathbf{a}_3$, s.t.

$$\begin{aligned} \mathbf{a}_1 &= (\cos\theta_1, \sin\theta_1, -r_1)^t, \\ \mathbf{a}_2 &= (\cos\theta_2, \sin\theta_2, -r_2)^t, \\ \mathbf{a}_3 &= (\cos\theta_3, \sin\theta_3, -r_3)^t, \end{aligned}$$

which are to be mapped by $\mathbf{P}$ to

$$\begin{aligned} \mathbf{e}_1 &= (\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, 0)^t = (0, 1, 0)^t, \\ \mathbf{e}_2 &= (\cos\varphi_1, \sin\varphi_1, 0)^t = (C_1, S_1, 0)^t, \\ \mathbf{e}_3 &= (\cos\varphi_2, \sin\varphi_2, -\sin|\theta_1 - \theta_3|)^t = (C_2, S_2, -\sin|\theta_1 - \theta_3|)^t. \end{aligned}$$

Note that φ_1 and φ_2 are fixed, though unknown. Also note that since we fixed the third component of $\mathbf{e}_3$ to be $-\sin|\theta_1 - \theta_3| < 0$, φ_2 ranges within $(-\frac{\pi}{2}, \frac{\pi}{2})$ (or, $C_2 > 0$). Our intention is to map the first basis line to x-axis; the intersection of the first basis line and the second basis line to the origin; the intersection of the first basis line and the third basis line to $(1, 0)^t$.

Since $ax + by + c = 0$ and $\lambda ax + \lambda by + \lambda c = 0$, $\lambda \neq 0$, represent the same line, we have

$$\begin{aligned} \lambda_1 \mathbf{P}\mathbf{a}_1 &= \mathbf{e}_1, \\ \lambda_2 \mathbf{P}\mathbf{a}_2 &= \mathbf{e}_2, \\ \lambda_3 \mathbf{P}\mathbf{a}_3 &= \mathbf{e}_3, \end{aligned}$$

where λ_1, λ_2 and λ_3 are non-zero constants. Equivalently,

$$\mathbf{P}(\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2, \lambda_3\mathbf{a}_3) = (\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3).$$

Then

$$\begin{aligned} (\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2, \lambda_3\mathbf{a}_3) &= \mathbf{P}^{-1}(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3) \\ &= \begin{pmatrix} -s\,\sin\phi & s\,\cos\phi C_1 - s\,\sin\phi S_1 & s\,\cos\phi C_2 - s\,\sin\phi S_2 \\ s\,\cos\phi & s\,\sin\phi C_1 + s\,\cos\phi S_1 & s\,\sin\phi C_2 + s\,\cos\phi S_2 \\ b_2 & b_1 C_1 + b_2 S_1 & b_1 C_2 + b_2 S_2 - \sin|\theta_1 - \theta_3| \end{pmatrix}. \end{aligned}$$

or

$$\mathbf{P} = \begin{pmatrix} -\sin\theta_1 & \cos\theta_1 & 0 \\ -\cos\theta_1 & -\sin\theta_1 & 0 \\ \csc(\theta_1-\theta_2)(r_2\sin\theta_1 - r_1\sin\theta_2) & \csc(\theta_1-\theta_2)(-r_2\cos\theta_1 + r_1\cos\theta_2) & 1 \end{pmatrix}.$$

Note that we have two solutions for $\mathbf{P}$ owing to the 180°-*rotation ambiguity.*

From section 3.3, the invariant $(\theta', r')^t$ of a line $(\theta, r)^t$ with respect to the basis lines can be obtained as follows:

$$(\theta', r')^t = \mathbf{H}((\theta, r)^t) = \mathbf{G}(\mathbf{P}(\cos\theta, \sin\theta, -r)^t) = \mathbf{G}(\mathbf{a}')$$

where

$$\mathbf{a}' = (-\sin(\theta-\theta_1), \cos(\theta-\theta_1), -r - r_2\csc(\theta_1-\theta_2)\sin(\theta-\theta_1) + r_1\csc(\theta_1-\theta_2)\sin(\theta-\theta_2))^t$$

or

$$\mathbf{a}' = (\sin(\theta-\theta_1), -\cos(\theta-\theta_1), -r - r_2\csc(\theta_1-\theta_2)\sin(\theta-\theta_1) + r_1\csc(\theta_1-\theta_2)\sin(\theta-\theta_2))^t,$$

depending upon which $\mathbf{P}$ is used, and hence

$$\begin{aligned} \theta' &= \theta - \theta_1 + \frac{\pi}{2}, \\ r' &= r + \csc(\theta_1-\theta_2)(r_2\sin(\theta-\theta_1) - r_1\sin(\theta-\theta_2)), \end{aligned}$$

or

$$\begin{aligned} \theta' &= \theta - \theta_1 + \frac{\pi}{2}, \\ r' &= -r - \csc(\theta_1-\theta_2)(r_2\sin(\theta-\theta_1) - r_1\sin(\theta-\theta_2)). \end{aligned}$$

We may store each encoded invariant (θ', r') redundantly in two entries of the hash table, (θ', r') and $(\theta', -r')$, during preprocessing. Then we may hit a match with either (θ', r') or $(\theta', -r')$ as the computed scene invaraint during recognition.

4.3 Line Invariants under Similarity Transformations

To determine a similarity transformation uniquely, we need a correspondence of a pair of triplets of lines, not all of which are parallel to each other or intersect at a common point.

Without loss of generality, we may assume that the first basis line intersects with the other two basis lines. One way to encode the fourth line in a similarity-invariant way is to find a transformation $\mathbf{P}$, which maps the first basis line to the x-axis, maps the second basis line to such that its intersection with the first basis line is the origin and maps the third basis line to such that its intersection with the first basis line is $(1, 0)^t$, then apply $\mathbf{P}$ to the fourth line.

A similarity transformation $\mathbf{T}$ in image space is of the form

$$\mathbf{T} = \begin{pmatrix} s\mathbf{R} & \mathbf{b} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} s\cos\phi & s\sin\phi & b_1 \\ -s\sin\phi & s\cos\phi & b_2 \\ 0 & 0 & 1 \end{pmatrix}$$

transformation $\mathbf{P}$ defined by

$$\mathbf{P} = (\mathbf{T}^{-1})^t = \begin{pmatrix} (\mathbf{R}^{-1})^t & 0 \\ -\mathbf{b}^t(\mathbf{R}^{-1})^t & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & 0 \\ -\mathbf{b}^t\mathbf{R} & 1 \end{pmatrix}, \tag{2}$$

so that

$$\mathbf{P}^{-1} = \mathbf{T}^t = \begin{pmatrix} \mathbf{R}^t & 0 \\ \mathbf{b}^t & 1 \end{pmatrix}.$$

Let a 2-D line be parameterized as $\cos\theta\, x + \sin\theta\, y - r = 0$ and let two additional basis lines be represented by their parameters $\mathbf{a}_1$ and $\mathbf{a}_2$, s.t.

$$\begin{aligned} \mathbf{a}_1 &= (\cos\theta_1, \sin\theta_1, -r_1)^t, \\ \mathbf{a}_2 &= (\cos\theta_2, \sin\theta_2, -r_2)^t, \end{aligned}$$

which are to be mapped by $\mathbf{P}$ to

$$\begin{aligned} \mathbf{e}_1 &= (\cos\frac{\pi}{2}, \sin\frac{\pi}{2}, 0)^t = (0,1,0)^t, \\ \mathbf{e}_2 &= (\cos\varphi, \sin\varphi, 0)^t = (C, S, 0)^t. \end{aligned}$$

Note that φ is fixed, though unknown. Our intention is to map the first basis line to x-axis (or, $y = 0$) and the intersection of the two basis lines to the origin.

Since $ax + by + c = 0$ and $\lambda ax + \lambda by + \lambda c = 0$, $\lambda \neq 0$, represent the same line, we have

$$\begin{aligned} \lambda_1 \mathbf{P}\mathbf{a}_1 &= \mathbf{e}_1, \\ \lambda_2 \mathbf{P}\mathbf{a}_2 &= \mathbf{e}_2, \end{aligned}$$

where λ_1 and λ_2 are non-zero constants. Equivalently,

$$\mathbf{P}(\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2) = (\mathbf{e}_1, \mathbf{e}_2).$$

Then

$$\begin{aligned} (\lambda_1\mathbf{a}_1, \lambda_2\mathbf{a}_2) &= \mathbf{P}^{-1}(\mathbf{e}_1, \mathbf{e}_2) \\ &= \begin{pmatrix} -\sin\phi & \cos\phi\, C - \sin\phi\, S \\ \cos\phi & \sin\phi\, C + \cos\phi\, S \\ b_2 & b_1 C + b_2 S \end{pmatrix}. \end{aligned}$$

Thus

$$\lambda_1 = \frac{-\sin\phi}{\cos\theta_1} = \frac{\cos\phi}{\sin\theta_1} = \frac{b_2}{-r_1}, \tag{3}$$

$$\lambda_2 = \frac{\cos\phi\, C - \sin\phi\, S}{\cos\theta_2} = \frac{\sin\phi\, C + \cos\phi\, S}{\sin\theta_2} = \frac{b_1 C + b_2 S}{-r_2}. \tag{4}$$

From (3) and (4), we may solve ϕ, b_1 and b_2. Substituting them in (2), we obtain

$$\mathbf{P} = \begin{pmatrix} \sin\theta_1 & -\cos\theta_1 & 0 \\ \cos\theta_1 & \sin\theta_1 & 0 \\ \csc(\theta_1 - \theta_2)(r_2\sin\theta_1 - r_1\sin\theta_2) & \csc(\theta_1 - \theta_2)(-r_2\cos\theta_1 + r_1\cos\theta_2) & 1 \end{pmatrix}$$

4 Line Invariants under Various Transformation Groups

Relevant transformations include rigid transformations, similarity transformations, affine transformations and projective transformations, depending on the manner in which an image is formed. Each of these classes of transformations forms a *group* and is a subgroup of the full group of projective transformations.

4.1 Encoding Lines by a Combination of Lines

To adapt the point geometric hashing technique to line features, we need to encode line features in terms of a combination of lines (as a basis) in a way invariant under transformations considered.

One way of encoding is to find a *canonical basis* and a *unique* transformation (in the transformation group under consideration) that maps a combination of lines (as a basis) to the canonical basis, then apply that transformation to the line to be encoded. For example, let $\mathbf{T}$ be a projective transformation that maps a basis b to $\mathbf{T}(b)$. If $\mathbf{P}$ is the unique transformation such that $\mathbf{P}(b) = b_c$ and $\mathbf{P}'$ is the unique transformation such that $\mathbf{P}'(\mathbf{T}(b)) = b_c$, where b_c is the chosen canonical basis, then we have $\mathbf{P} = \mathbf{P}' \circ \mathbf{T}$ and any other line l and its correspondence $l' = \mathbf{T}(l)$ will be mapped to $\mathbf{P}(l)$ by $\mathbf{P}$ and $\mathbf{P}'(l')$ by $\mathbf{P}'$ respectively. Note that $\mathbf{P}'(l') = \mathbf{P}'(\mathbf{T}(l)) = \mathbf{P}' \circ \mathbf{T}(l) = \mathbf{P}(l)$ and conclude that this is an invariant encoding[3].

Various subgroups of the projective transformation group require different bases. We will discuss them in the following sections.

Note that throughout the following discussion, when a line is represented by its normal parameterization (θ, r), we restrict θ to be in $[0, \pi)$ and r to be in $\mathbf{R}$ and if the computed invariant (θ', r') has θ' not in $[0..\pi)$, we adjust it by adding π or $-\pi$ and adjust the value of r' by flipping its sign accordingly.

4.2 Line Invariants under Rigid Transformations

To determine a rigid transformation uniquely, we need a correspondence of two ordered *triplets* of lines, not all of which are parallel to each other and intersect at a common point. A correspondence of two ordered *pairs* of non-parallel lines is not sufficient to determine a rigid transformation uniquely. However, it determines a rigid transformation *up to a* $180°$*-rotation ambiguity* and this ambiguity can be broken if we know the position of a third line.

One way to encode the third line in terms of a pair of non-parallel lines in a rigid-invariant way (*up to a* $180°$*-rotation ambiguity*) is to find a transformation $\mathbf{P}$, which maps the first basis line to the x-axis and maps the second basis line to such that its intersection with the first basis line is the origin, then apply $\mathbf{P}$ to the third line.

A rigid transformation $\mathbf{T}$ in image space is of the form

$$\mathbf{T} = \begin{pmatrix} \mathbf{R} & \mathbf{b} \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} \cos\phi & \sin\phi & b_1 \\ -\sin\phi & \cos\phi & b_2 \\ 0 & 0 & 1 \end{pmatrix},$$

where $\mathbf{R}$ is the rotation matrix and $\mathbf{b}$, the translation vector. This corresponds to the parameter-space where r is the perpendicular distance of the line to the origin and θ is the angle between a normal to the line and the positive x-axis.

This unique parameterization of lines relates to the preceding parameterization of lines. Let $\mathbf{F} : \mathbf{R}^2 \mapsto \mathbf{R}^3$ be a mapping such that

$$\mathbf{F}((\theta, r)^t) = (\cos\theta, \sin\theta, -r)^t$$

If we restrict the domain of θ to be in $[0, \pi)$, then $\mathbf{F}^{-1}$ exists. Define another mapping $\mathbf{G} : \mathbf{R}^3 \mapsto \mathbf{R}^2$ by $\mathbf{F}^{-1}$ as

$$\mathbf{G}((a_1, a_2, a_3)^t) = \begin{cases} \mathbf{F}^{-1}((\frac{a_1}{\sqrt{a_1^2+a_2^2}}, \frac{a_2}{\sqrt{a_1^2+a_2^2}}, \frac{a_3}{\sqrt{a_1^2+a_2^2}})^t) & \text{if } a_2 \geq 0 \\ \mathbf{F}^{-1}((\frac{-a_1}{\sqrt{a_1^2+a_2^2}}, \frac{-a_2}{\sqrt{a_1^2+a_2^2}}, \frac{-a_3}{\sqrt{a_1^2+a_2^2}})^t) & \text{otherwise} \end{cases}$$

where a_1 and a_2 are not both equal to 0. Then

$$\mathbf{G}((a_1, a_2, a_3)^t) = \begin{cases} (\tan^{-1}(\frac{a_2}{a_1}), \frac{-a_3}{\sqrt{a_1^2+a_2^2}})^t & \text{if } a_2 > 0 \\ (0, \frac{-a_3}{a_1})^t & \text{if } a_2 = 0 \\ (\tan^{-1}(\frac{-a_2}{-a_1}), \frac{a_3}{\sqrt{a_1^2+a_2^2}})^t & \text{if } a_2 < 0 \end{cases}$$

where the range of $\tan^{-1}$ is in $[0..\pi)$ and $\tan^{-1}(\infty) = \frac{\pi}{2}$.

Then $\mathbf{G}$ maps a point $\mathbf{a} = (a_1, a_2, a_3)^t$ in parameter space, where a_1 and a_2 are not both equal to 0, to $(\theta, r)^t = \mathbf{G}(\mathbf{a})$ in (θ, r)-space such that

$$a_1 w_1 + a_2 w_2 + a_3 w_3 = 0$$

and

$$\cos\theta\, w_1 + \sin\theta\, w_2 - r\, w_3 = 0$$

define the same line.

Let $(\theta, r)^t$ be the (θ, r)-parameter defining a line (in fact, the line $\cos\theta\, x + \sin\theta\, y = r$). Then $\mathbf{F}((\theta, r)^t) = \mathbf{a}$, where $\mathbf{a} = (\cos\theta, \sin\theta, -r)^t$, is a point in parameter space. A transformation $\mathbf{P}$ changes the coordinate of $\mathbf{a}$ to $\mathbf{a}' = \mathbf{P}\mathbf{a}$ in parameter space. Substituting $\mathbf{F}((\theta, r)^t)$ for $\mathbf{a}$ in $\mathbf{a}' = \mathbf{P}\mathbf{a}$, we get

$$\mathbf{a}' = \mathbf{P}\mathbf{F}((\theta, r)^t)$$

and hence $(\theta', r')^t = \mathbf{G}(\mathbf{a}') = \mathbf{G}(\mathbf{P}\mathbf{F}((\theta, r)^t))$ defines the same line as $\mathbf{a}'$ (or $\lambda\mathbf{a}'$, $\lambda \neq 0$).

Thus a transformation of a point in parameter space results in the transformation of $(\theta, r)^t$ in (θ, r)-space. The change of coordinate of $(\theta, r)^t$ in (θ, r)-space is given by

$$(\theta', r')^t = \mathbf{H}((\theta, r)^t) \tag{1}$$

where $\mathbf{H} = \mathbf{G} \circ \mathbf{P} \circ \mathbf{F}$.[1]

[1] We abused the notation by using $\mathbf{P}(\mathbf{v})$ to denote $\mathbf{P}\mathbf{v}$ (matrix $\mathbf{P}$ multiplies vector $\mathbf{v}$). We will continue to use $\mathbf{P}(\mathbf{v})$ or $\mathbf{P}\mathbf{v}$ interchangeably when no ambiguity occurs.

A point $(x, y)^t$ in image space is represented by a non-zero 3-D point $\mathbf{w} = (w_1, w_2, w_3)^t$ in homogeneous coordinate systems, such that

$$x = \frac{w_1}{w_3} \quad \text{and} \quad y = \frac{w_2}{w_3},$$

where $w_3 \neq 0$. This representation is *not* unique, since $\lambda\mathbf{w}$, for any $\lambda \neq 0$ is also a homogeneous representation of $(x, y)^t$.

Every non-singular 3×3 matrix defines a 2-D projective transformation of homogeneous coordinates. Various subgroups of the projective transformation group are defined by different restrictions on the form of the matrix $\mathbf{T}$.

3.2 Change of Coordinates in Parameter Space

A line in a 2-D plane can be represented by its *parameter vector* and is usually parameterized as

$$a_1 w_1 + a_2 w_2 + a_3 w_3 = 0$$

or

$$\mathbf{a}^t \mathbf{w} = 0,$$

where $\mathbf{a} = (a_1, a_2, a_3)^t$ is the parameter vector of the line (note that a_1 and a_2 are not both equal to 0) and $\mathbf{w} = (w_1, w_2, w_3)^t$ is a homogeneous coordinate of any point on the line. This representation is *not* unique, since $\lambda\mathbf{a}$, for any $\lambda \neq 0$, is also a representation of the same line. Herein, parameter vector $\mathbf{a}$ defines a point in 3-D *parameter space*.

A transformation $\mathbf{T}$ in image space changes the coordinate of every point $\mathbf{w}$ on a line to $\mathbf{w}'$ by $\mathbf{w}' = \mathbf{T}\mathbf{w}$. Substituting $\mathbf{w} = \mathbf{T}^{-1}\mathbf{w}'$ in $\mathbf{a}^t\mathbf{w} = 0$, we get

$$\mathbf{a}^t \mathbf{T}^{-1} \mathbf{w}' = 0$$

and hence

$$((\mathbf{T}^{-1})^t \mathbf{a})^t \mathbf{w}' = 0,$$

or

$$\mathbf{a}'^t \mathbf{w}' = 0, \text{ where } \mathbf{a}' = (\mathbf{T}^{-1})^t \mathbf{a}.$$

This shows that the change of the coordinate of the point in 3-D parameter space is given by

$$\mathbf{a}' = \mathbf{P}\mathbf{a}.$$

where $\mathbf{P} = (\mathbf{T}^{-1})^t$.

3.3 Change of Coordinates in (θ, r) Space

Line features of an image are usually extracted by the *Hough* transform. A common implementation of the *Hough* transform applies a *normal parameterization* suggested by Duda and Hart[1], in the form

$$\cos\theta\, x + \sin\theta\, y = r,$$

(i) compute the invariants of all the remaining points in terms of the basis b;

(ii) use the computed invariants to index the 2-D hash table entries, in each of which we record a node (M, b).

Note that all feasible bases have to be used in forming the set of invariants to be stored. In particular, all the permutations (up to $k!$) of the k points needed to calculate the invariants have to be considered.

The recognition stage
Given a scene containing n feature points,

(i) choose a feasible set b of k points;

(ii) compute the invariants of all the remaining points in terms of this basis b;

(iii) use each computed invariant to index the 2-D hash table and *hit* all (M_i, b_j)'s that are stored in the entries retrieved;

(iv) histogram all (M_i, b_j)'s with the number of *hits* received;

(v) assume the existence of an instance of model M_i in the scene, if (M_i, b_j), for some j, peaks in the histogram with sufficiently many *hits*;

(vi) repeat from step **(i)**, if all hypotheses established in step **(v)** fail verification.

3 Change of Coordinates in Various Spaces

The idea behind the geometric hashing method is to encode local geometric features in a manner which is invariant under the geometric transformation that model objects undergo during formation of the class of images being analyzed. This encoding can then be used as a hash function that makes possible fast retrieval of model features from the hash table, which can be viewed as an encoded model base. The same technique can be applied directly to line features, without resorting to point features indirectly derived from lines, if we use any method of encoding line features in a way invariant under transformations considered.

3.1 Change of Coordinates in Image Space

It is often easiest to deal with homogeneous coordinates, since this makes projective transformations *linear* and easily subject to matrix operations.

A change of coordinates in homogeneous coordinates is given by

$$\mathbf{w}' = \mathbf{T}\mathbf{w},$$

where $\mathbf{w} = (w_1, w_2, w_3)^t$ is the homogeneous coordinate before transformation, $\mathbf{w}' = (w_1', w_2', w_3')^t$ is the homogeneous coordinate after transformation and $\mathbf{T}$ is a non-singular 3×3 matrix. Note that $\mathbf{T}$ and $\lambda\mathbf{T}$, for any $\lambda \neq 0$, define the same transformation, since we are dealing with homogeneous coordinates.

1 Introduction

Geometric Hashing[5][6] is a model-based object recognition approach based on precompiling redundant transformation-invariant information derived for object models into a hash table, and using the invariants computed from a scene for fast indexing into the hash table to hypothesize possible matches between object instances and object models during recognition. Since only local geometric features are used to compute the invariants, it can cope well with the problems caused by partial overlapping and occlusion.

In a noisy image, point feature locations are inherently error-prone and the analysis of geometric hashing on point sets [2] shows considerable sensitivity to noise. In contrast, line features are more robust and can be extracted by the *Hough* transform method with greater accuracy, since more image points are involved. Although point features can be obtained by intersection of lines, if this is done, the uncertainty (noise) of lines accumulates and contributes to the uncertainty of the point positions thus derived. Hence it can be more advantageous to work directly with lines by computing the geometric invariants of lines directly from a combination of lines.

We examine line invariants under various transformations for the recognition of 2-D (or, flat 3-D) objects which are modeled by lines.

We organize this paper as follows. In section 2, we give a brief review of the original description of the geometric hashing method to make the paper self-contained. Section 3 examines the way in which coordinate changes act on various geometric spaces of potential interest for recognition. This section is preliminary to its following section. Section 4 describes a method to encode line features in a transformation invariant way and gives the derivation of that encoding. Section 5 discusses best least-squares match procedures for transformed models and their instances in a scene. Section 6 outlines some future directions.

2 Review of the Geometric Hashing Method

This section reviews the geometric hashing method. For more detail, refer to [7].

Geometric hashing involves two stages: a preprocessing stage and a recognition stage. In the preprocessing stage, we construct a model representation by computing and storing redundant, transformation-invariant model information in a hash table. Then, during the recognition stage, the same invariants are computed from features in a scene and used as indexing keys to retrieve from the hash table the possible matches with the model features. If a model's features scores enough *hits*, we hypothesize the existence of an instance of that model in the scene.

The pre-processing stage
Models are processed one by one. New models added to the model base can be processed and encoded into the hash table independently.
We proceed as follows: For each model M and for every feasible basis b, consisting of k points (k will depend on the transformations the model objects undergo during formation of the class of images to be analyzed),

Using Line Invariants for Object Recognition by Geometric Hashing

Frank Chee-Da Tsai

Robotics Research Laboratory,
Courant Institute of Mathematical Sciences,
New York University
715 Broadway, 12FL
New York, N.Y. 10003
tsai@robust.nyu.edu

Technical Report No. 0625

Prepared January 1993; Revised February 1993

Abstract

Geometric Hashing is a model-based object recognition technique for detecting objects which can be partially overlapping or partly occluded. It precompiles, from local geometric features, redundant transformation-invariant information of the models in a hash table and uses the invariants computed from a scene for fast indexing into the hash table to hypothesize possible matches between object instances and object models during recognition.

In its simplest form, the geometric hashing method assumes relatively noise-free data and is applied to objects with points as local features. However, extracting of the locations of point features is inherently error-prone and the analysis of geometric hashing on point sets shows considerable noise sensitivity. Line features can generally be extracted with greater accuracy.

We investigate the use of line features for geometric hashing applied to 2-D (or flat 3-D) object recognition and derive, from a combination of line features, invariants for lines under various geometric transformations.