Localization and navigation using projective invariants

Raman Arora

Camera Network Study Group
July 18th, 2007

Outline

- Localization of robot using an omni- directional camera by geometric matching and 1-D projective invariant matching (Marhic et. al. 1998).

- Localization of robot with a single camera using 2-D planar projective invariants (Roh et. al. 1997)

- Temporal calibration of video sequences from unsynchronized cameras using 2-D projective invariants (Velipasalar 2005).

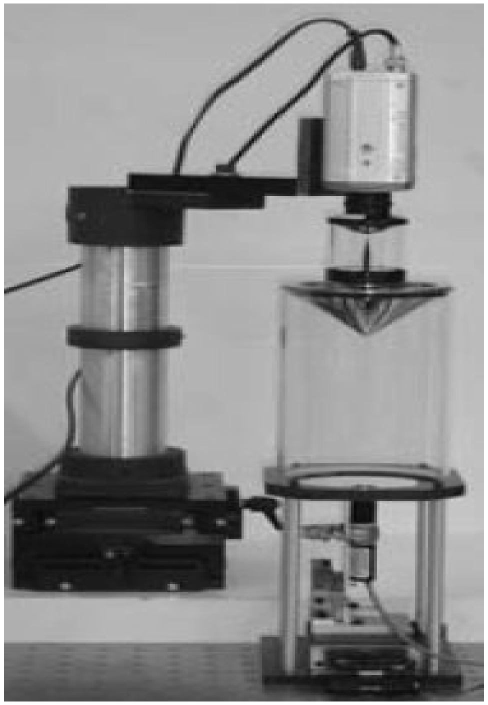
- Landmark based navigation of a robot using projective invariants (Tsonis et. al. 1998).

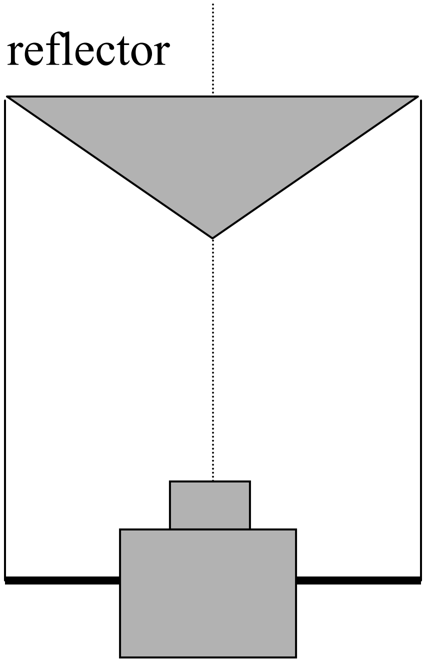
Possible Methodologies for localization

- Active beacons
  - Ultrasonic ranging sensor
    - Little processing but large uncertainty on real target point. Therefore, needs a corrective method.
  - Laser sensors
    - Produce narrow range, more accurate.
- Vision based methods
  - Stereoscopic – multiple cameras to capture panoramic scene.
  - Catadioptric – single camera with a conic or parabolic reflector.
- Proprioceptive sensors
  - Dead-reckoning.

Omni-directional sensor (Marhic et. al. 1998)

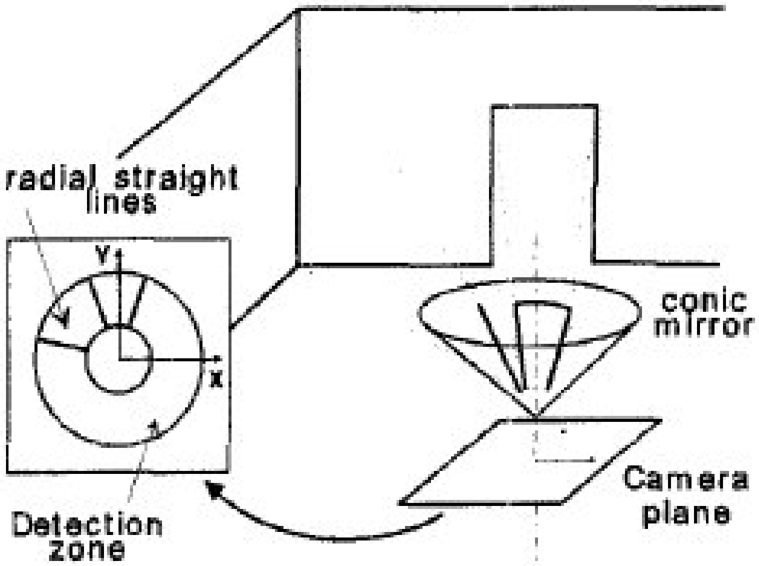
- Catadioptric imaging
  - Conic reflector vertically oriented
  - Single static camera





Omni-directional sensor (Marhic et. al. 1998)

- Vertical lines are usually the most distinctive or contrasted feature both indoors and outdoors.
- Vertical lines are projected as radial lines passing through the apex of conic reflector



Processing

- Extracting radius lines (Feature detection)
  - Deepen parts of the scene representing radius lines
  - Find characteristic parameters for line detection.
- Matching of surrounding recorded marks and observed scene (Feature matching)

Detecting radial lines

Possible approaches:

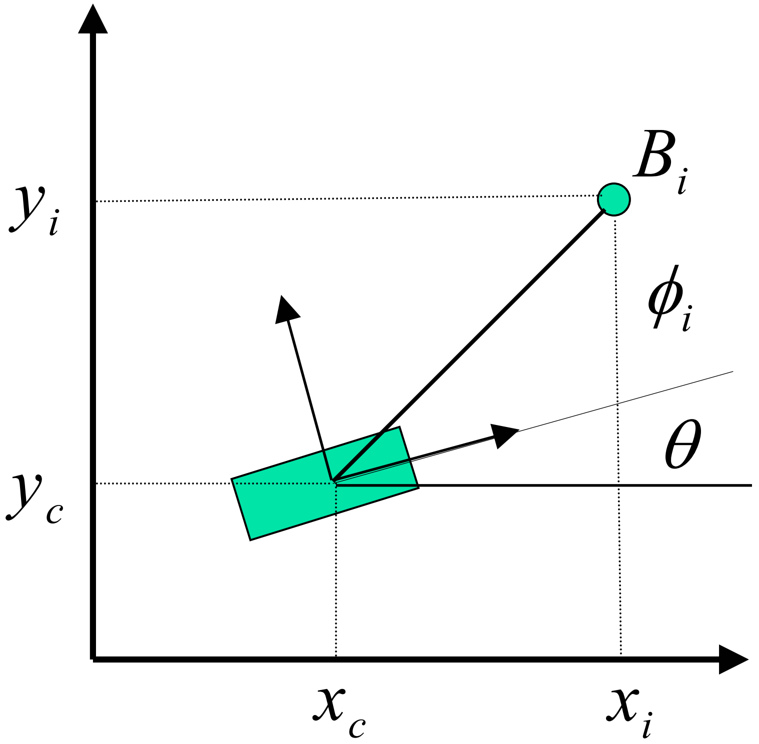
- Hough transform (finding pixels belonging to same line).
- Group pixels in areas, compute grey level gradients for every pixel, group directional gradients, find connected components.
- Attractive areas research:
  - Find grey levels on concentric circles with the apex of cone as the centre.
  - Use Sobel operator to get contrast on the circles.
  - Identify high gradient points (crossing pts for radius lines).
  - Group points belonging to the same lines

Localization

- Relating real and observed world
  - Need to find the three attitude parameters (x_c, y_c, θ) of the robot.
  - Atleast three radius lines are necessary to solve the set of relations

$$\tan(\theta + \phi_i) = \frac{y_i - y_c}{x_i - x_c}$$

  - Numerical methods may be employed to solve the equations above.
- Matching
  - For each set of three radius lines, find the solution and see which solution matches most other beacons.



Using projective invariant: 1-D cross ratio

- Method outlined above suffers from parasite straight lines.
- Cross ratio may be employed to resolve the matching between the model and omni-directional image.
- Matching with cross-ratio does not require calibration.

1-D Cross-ratio

- Cross-ratio is the most fundamental projective invariant and all other projective invariants can be derived from it.
- ***Definition***: For any four collinear points $P_1,\ldots,P_4$ the cross-ratio is defined as

$$\rho = \frac{D_{13}D_{24}}{D_{14}D_{23}}$$

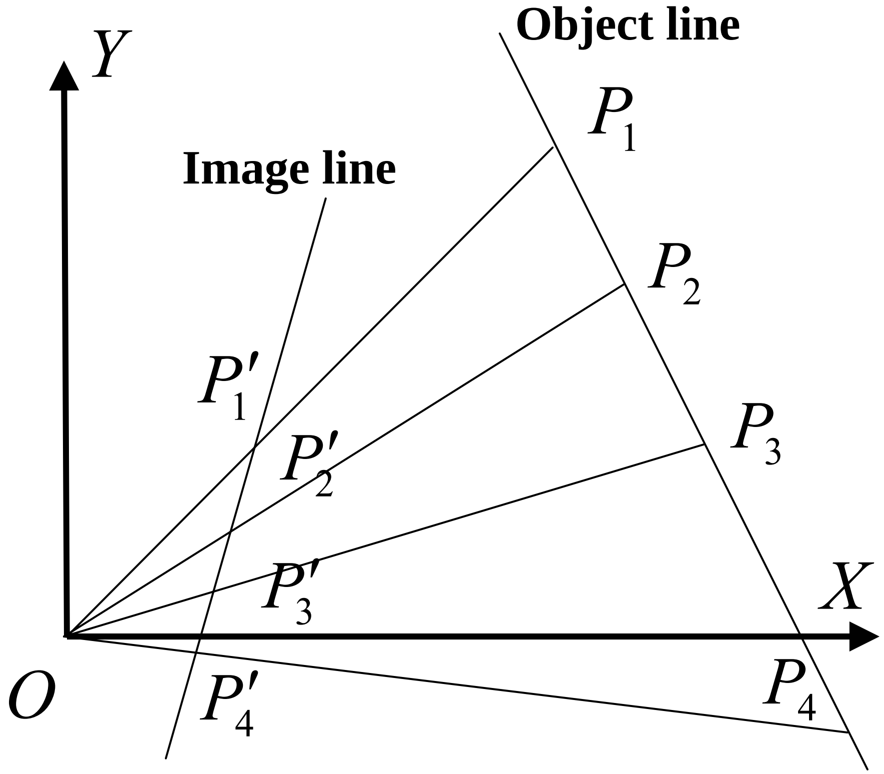
where D_{ij} is the distance between P_i and P_j

Cross-ratio

- Theorem: The cross-ratio of distances between any four points in the object line is the same as the cross-ratio of distances between their images in any image line,

$$\rho = \frac{D_{13}D_{24}}{D_{14}D_{23}} = \frac{D'_{13}D'_{24}}{D'_{14}D'_{23}}$$

where D'_{ij} is the distance between P'_i and P'_j.

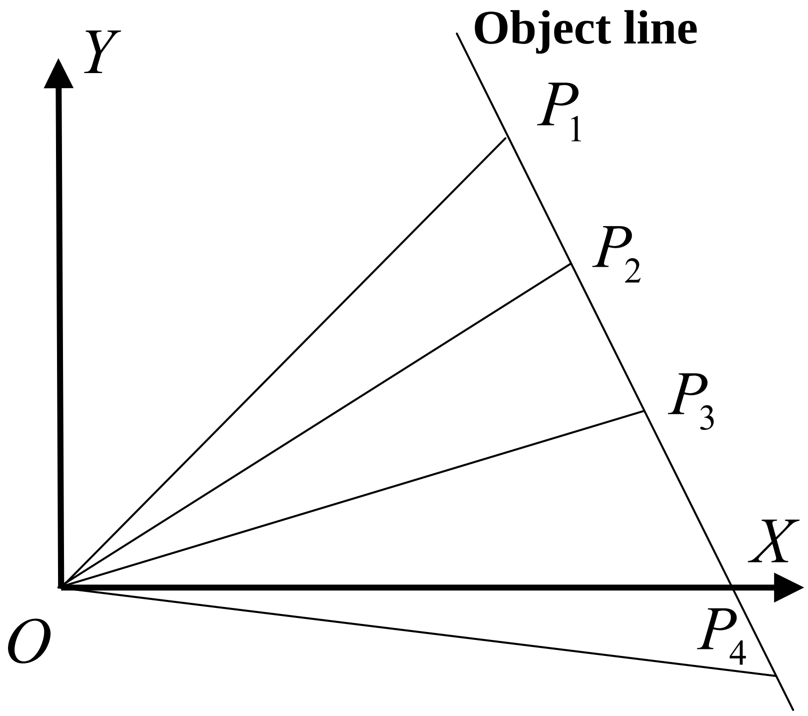


Dual of cross-ratio

- Since points and lines are dual (dual relation to collinearity being coincident), cross ratio for a pencil of four lines is defined as

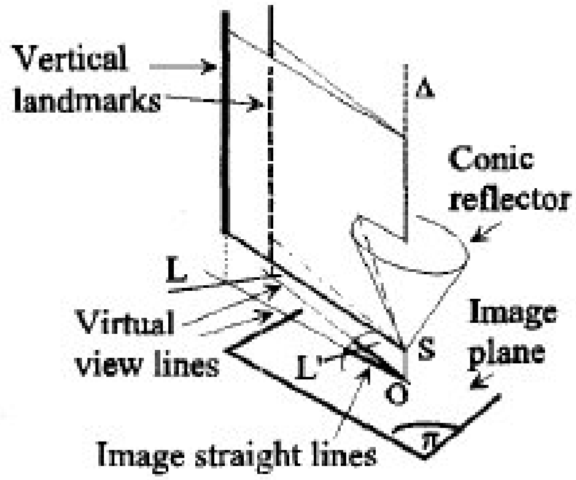
$$\rho = \frac{\sin\alpha_{13}\sin\alpha_{24}}{\sin\alpha_{14}\sin\alpha_{23}}$$

where α_{ij} is the angle subtended at the point of incidence by the line segment P_iP_j



Cross-ratio by radial lines

- Line L′ is the projective image of the line L.
- The coordinates of points on L and L ′ can be related by a 2x2 matrix T, x ′ = Tx.
- The matrix T has three essential parameters since the scale is not important.



- Theorem: Any homography preserves cross-ratio.

Numeration problem

- Cross-ratio depends on the order in which points are marked.
- Out of 24 possible permutations of four points, only 6 give different values for cross-ratios,

$$\begin{array}{lll} \rho_1 = \rho & \rho_3 = 1 - \rho_1 & \rho_5 = -\rho_1 \rho_4 \\ \rho_2 = \rho_1^{-1} & \rho_4 = \rho_3^{-1} & \rho_6 = -\rho_2 \rho_3 \end{array}$$

- Symmetric functions that are invariant to permutations may be used to combine the six cross-ratios, for instance

$$I_1 = \sum_{i=1}^{6} \rho_i$$

- A preferred permutation invariant is $I_2 = \dfrac{\left(\rho^2 - \rho + 1\right)^3}{\left(\rho^2 - \rho\right)^2}$

Plane projective invariants (Roh et. Al. 1997)

- ***Definition***: Given five points $p_{1,\ldots,}p_5$ on the projective plane, no three of which are collinear, two independent projective invariants are defined as

$$I_1 = \frac{|M_{421}||M_{532}|}{|M_{432}||M_{521}|}, I_2 = \frac{|M_{421}||M_{531}|}{|M_{431}||M_{521}|}$$

where $|M_{abc}|, \{a,b,c\} \in \{1,\ldots,5\}$ denotes the determinant of the matrix M_{abc} whose columns are the homogenous coordinates of the points $p_{a,}p_b$ and p_c.

Localization and obstacle detection (Roh 97)

- A method using cross-ratio and plane projective invariants is given in Roh et. al. for localization and obstacle detection while navigating in corridors and similar indoor environments.
- Its assumed that robot’s environment has flat ground plane and two sidelines are formed by floor and two sidewalls.
- The environmental map database is assumed to be available for matching between model and the scene.
- Intersection points between floor and the vertical lines of door frames are used as point features to compute cross-ratios.

Localization and obstacle detection (Roh 97)

- A database of pre-computed cross ratios of point features is constructed and used for finding correspondence between model and the scene.
- The locations of obstacles inside the risk zone are also computed the same way.
- If p_i and P_i, $i = 1,\ldots,5$ represent the coordinates of points on the image plane and the corresponding points in the object plane respectively, then

$$I_1 = \frac{\left|[p_4 p_2 p_1]\right| \left|[p_5 p_3 p_2]\right|}{\left|[p_4 p_3 p_2]\right| \left|[p_5 p_2 p_1]\right|} = \frac{\left|[P_4 P_2 P_1]\right| \left|[P_5 P_3 P_2]\right|}{\left|[P_4 P_3 P_2]\right| \left|[P_5 P_2 P_1]\right|}$$

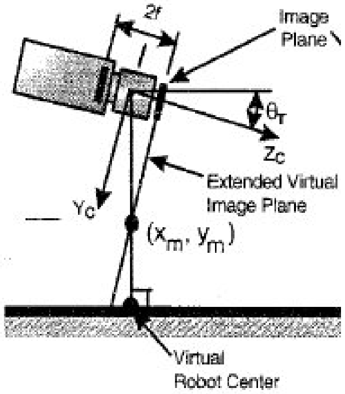
$$I_2 = \frac{\left|[p_4 p_2 p_1]\right| \left|[p_5 p_3 p_1]\right|}{\left|[p_4 p_3 p_1]\right| \left|[p_5 p_2 p_1]\right|} = \frac{\left|[P_4 P_2 P_1]\right| \left|[P_5 P_3 P_1]\right|}{\left|[P_4 P_3 P_1]\right| \left|[P_5 P_2 P_1]\right|}$$

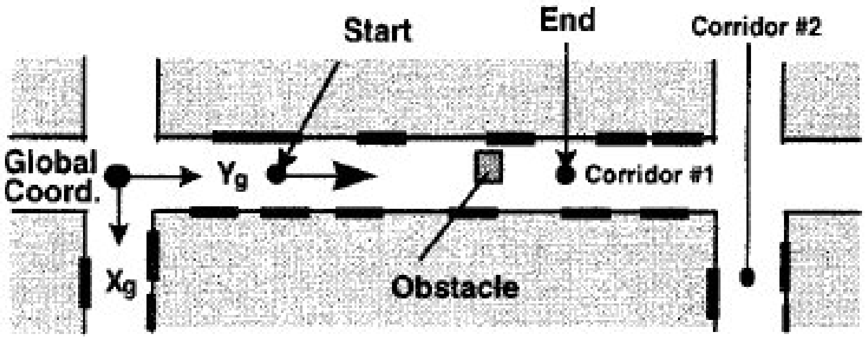
Localization and obstacle detection (Roh 97)

- The two equations above can be solved uniquely for localization: In order to find the relative position (X_5, Y_5) of an object point with respect to known four points $(X_1, Y_1), (X_2, Y_2), (X_3, Y_3)$ and (X_4, Y_4), (having found the image coordinates of the five points), the following system of equations can be solved,
$$AX_5 - BY_5 = -C$$
$$DX_5 - EY_5 = -F$$
where A, B, C, D, E and F can be expressed in terms of the invariants I_1 and I_2 and known coordinates.

Localization and obstacle detection (Roh 97)

- If the fifth point corresponds to the robot center, we get the localization. If it corresponds to an unexpected object on the risk zone, we get obstacle detection.





Temporal calibration of multiple video sequences (Velipasalar et. al. 2005)

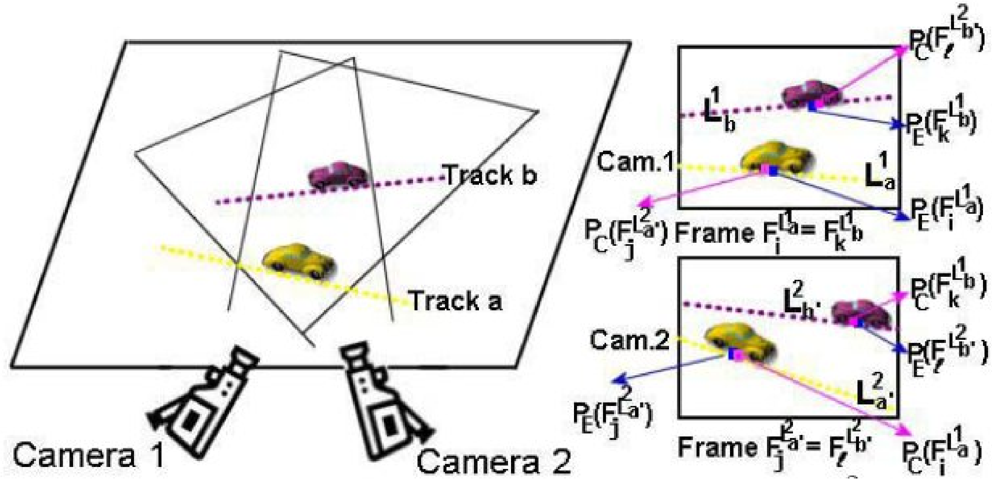
- Multi-camera systems receive increasing interest these days since single camera provides only a *limited field of view* and several applications (like surveillance) require larger coverage areas and longer tracking times. Another problem with single camera is that of *occlusion*.
- *Temporal calibration* identifies corresponding frames in video sequences captured by different cameras and is very important for multi-camera systems.
- Calibration using a synchronous master clock is expensive.
- Velipasalar et. al. present an image processing based method for temporal calibration from unsynchronized cameras.

Overview of the algorithm (Velipasalar 2005)

- Track each foreground object, extracting its location in the current sequence and finding the corresponding location in the other sequence using *projective invariants*.
- Find matching tracks in the video sequences and recovering an initial frame offset value for the match.
- Perform a confidence check for each matched track pair by using the recovered offset to find the most reliable matching track pair and candidate offset.
- Assumptions:
  - the cameras are static and have the same frame rate;
  - objects move on a planar surface and bottom parts of objects are visible, although briefly.

Operation scenario (Velipasalar 2005)

- L_a^c denotes the label of the a^{th} track in the c^{th} camera view, $c \in \{1,2\}, a \in \{1,\ldots,N_c\}$ where N_c is the number of tracks.
- $F_i^{L_a^c}$ is the frame number for the i^{th} point in the track L_a^c .



- The frame offset is $F_j^{L_{a'}^2} - F_i^{L_a^1}$ where a' is the track in sequence captured by camera 2 corresponding to track a in the sequence captured by camera 1.

Computing corresponding locations (Velipasalar 05)

- Denote the two cameras by C^i and C^j and a point on the ground plane of C^j by $p_g^{(j)}$. The corresponding location $p_g^{(i)}$ in the view of C^i is computed using projective invariants,

$$I_1 = \frac{\left|M^{(1)}_{421}\right|\left|M^{(1)}_{532}\right|}{\left|M^{(1)}_{432}\right|\left|M^{(1)}_{521}\right|} = \frac{\left|M^{(2)}_{421}\right|\left|M^{(2)}_{532}\right|}{\left|M^{(2)}_{432}\right|\left|M^{(2)}_{521}\right|}$$

$$I_2 = \frac{\left|M^{(1)}_{421}\right|\left|M^{(1)}_{531}\right|}{\left|M^{(1)}_{431}\right|\left|M^{(1)}_{521}\right|} = \frac{\left|M^{(2)}_{421}\right|\left|M^{(2)}_{531}\right|}{\left|M^{(2)}_{431}\right|\left|M^{(2)}_{521}\right|}$$

- Four pairs of corresponding points in the views of C^i and C^j are chosen offline on the ground plane. Then for any fifth point in the view of C^i, the corresponding point in the view of C^j can be found using the invariants.

Matching the tracks (Velipasalar 2005)

- A track is stored as a sequence

$$L_a^c \longrightarrow \left\{ \left(F_1^{L_a^c}, P_E(F_1^{L_a^c}), P_c(F_1^{L_a^c})\right) \cdots \left(F_n^{L_a^c}, P_E(F_n^{L_a^c}), P_c(F_n^{L_a^c})\right) \right\}$$

  where $P_E(F_1^{L_a^c}) = (x_{E_i}^{L_a^c}, y_{E_i}^{L_a^c})$ is the extracted location of the foreground object in the current view and $P_C(F_1^{L_a^c}) = (x_{C_i}^{L_a^c}, y_{c_i}^{L_a^c})$ is the corresponding location of $P_E(F_1^{L_a^c})$ in the other view.

- The distance between points of tracks in different cameras

$$D\left(F_i^{L_a^1}, F_j^{L_t^2}\right) = d\left(P_c(F_i^{L_a^1}), P_E(F_j^{L_t^2})\right) + d\left(P_E(F_i^{L_a^1}), P_C(F_j^{L_t^2})\right)$$

- The track matching problem

$$\left\{t^*, i^*, j^*\right\} = \underset{\substack{t \in \{1,\ldots,N_2\} \\ i \in \{1,\ldots,|L_a^1|\} \\ j \in \{1,\ldots,|L_t^2|\}}}{\arg\min} \left\lfloor D(F_i^{L_a^1}, F_j^{L_t^2}) + D(F_{i+\Delta}^{L_a^1}, F_{j+\Delta}^{L_t^2}) \right\rfloor$$

where Δ is the frame offset.

Landmark-based navigation using projective invariants (Tsonis et. al. 1998)

- The 2-D cross-ratio is used to recognize and store landmarks during a learning phase.
- The stored landmarks are matched to re-discovered landmarks at navigation time.
- Instead of using pre-designed engineered landmarks or selected landmarks like straight-lines, the approach presented in this paper addresses the problem in more general and realistic workspaces.
- The landmarks derived from the captured images have to satisfy some *saliency* and *spatial dispersion*.
- It is assumed that robot's environment contains planar surfaces.

Learning phase: Permutation insensitive 2-D projective invariant (Tsonis et. al. 1998)

- Two-dimensional cross-ratio is permutation sensitive
- Any quintuple gives five different values for the 2-D cross-ratio depending on the order.
- However, any two of the five different cross-ratios can determine the other three.

$$\mu = [P_1, P_2, P_3, P_4, P_5] = \frac{\left|[P_1 P_2 P_4]\right| \left|[P_1 P_3 P_5]\right|}{\left|[P_1 P_3 P_4]\right| \left|[P_1 P_2 P_5]\right|}, \qquad \nu = [P_2, P_1, P_3, P_4, P_5]$$

- A permutation sensitive 2-D projective invariant

$$K(\mu, \nu) = J(\mu) + J(\nu) + J(\frac{\mu}{\nu}) + J(\frac{\nu - 1}{\mu - 1}) + J(\frac{\mu(\nu - 1)}{\nu(\mu - 1)})$$

where

$$J(\lambda) = \frac{2\lambda^6 - 6\lambda^5 + 9\lambda^4 - 8\lambda^3 + 9\lambda^2 - 6\lambda + 2}{\lambda^6 - 3\lambda^5 + 3\lambda^4 - \lambda^3 + 3\lambda^2 - 3\lambda + 1}$$

Learning phase: Visual landmarks (Tsonis 98)

- Visual landmarks are defined to be the sets containing sub-landmarks.
- Sub-landmarks are *quintuples* of coplanar points derived by
  - first using a robust corner detector (the potential landmarks form *corner map*)
  - constructing a *saliency map* comprising of points that form distinct enough patterns; using features like area correlation, image entropy in neighborhoods.
  - choosing points that are close enough but satisfy a spatial dispersion threshold.
  - checking for co-planarity of points
    - By identifying corresponding quintuples in consecutive frames using a covariance test.
    - verifying the permutation insensitive projective invariants for quintuples in consecutive frames.
- Topological map construction: storing the projective invariant with each sub-landmark, along with references to navigational preferences.

Landmark Recognition (Tsonis 98)

- Follows the same procedure for extracting the landmarks as during the learning phase.
- The projective invariants for quintuples located in the scene are compared with stored values to find correspondence.

References

- B. Marhic, E. M. Mouaddib and C. Pegard, “A localisation method with an omnidirectional vision sensor using projective invariant,” Proc. Int'l conf. Intelligent Robots and Systems, 1998, pp. 1078-1083.
- L. Delahoche, C. Pegard, B. Marhic and P. Vasseur, “A navigation system based on an omnidirectional vision sensor,” Proc. Int'l conf. IROS 1997, pp. 718-724.
- C. Pegard and E. M. Mouaddib, “A mobile robot using a panoramic view,” Proc. Int'l conf. on Robotics and Automation, 1996, pp. 89-94.
- K. S. Roh, Wang Heon Lee, and In So Kweon, “Obstacle detection and self-localization without camera calibration using projective invariants,” Proc. Intelligent Robots and Systems (IROS'97), 1997, pp. 1030-1035.
- S. Velipasalar, W. Wolf, “Frame-level temporal calibration of video sequences from unsynchronized cameras by using projective invariants,” Proc. Advanced Video Signal-based Surveillance (AVSS'05), 2005, pp. 462-467.
- V. S. Tsonis, K.V. Chandrinos; Trahanias, P.E., “Landmark-based navigation using projective invariants,” Proc. Int'l conf. IROS 1998, pp. 342-347.