



## Object recognition in clutter image using putative feature point matching algorithm

Ketaki P Dahale

Gokhale College of Engineering, Nashik, India.

### ARTICLE INFO

#### Article history:

Received: 16 April 2014;

Received in revised form:

21 August 2014;

Accepted: 29 August 2014;

#### Keywords

Azomethine ylide,  
Diels-Alder cycloaddition,  
Enantioselective catalysis,  
Microwave-assisted synthesis.

### ABSTRACT

The goal of this project is to automatically detect and recognize some objects in an image by using a multi-agents architecture. A knowledge database is thus necessary and should contain an invariant description of known objects for the desired application. Point matching is an important aim in the research work. A digital image may undergo any arbitrary translational, rotational changes because of which the object shape may change. In this paper we present a novel approach for shape and feature matching using putative point matching.

© 2014 Elixir All rights reserved

### Introduction

In many applications of computer vision, pattern recognition and medical image analysis, one common procedure is to match two or more point sets, and no rigid point set matching is particularly difficult because the possible no rigid deformation of the model shape is numerous. In practice, the scene is often contaminated by clutters, making the point set matching problem more complicated. In this paper, we focus on how to locate a deformable shape in cluttered scenes under the no rigid point set matching framework. The shape may undergo arbitrary translational and rotational changes, and it may be no rigidly deformed and corrupted by clutters.

To address the problem of rotation invariant no rigid point set matching, they proposed two methods for shape representation. The shape context (SC) feature descriptor was used and we constructed graphs on point sets where edges are used to determine the orientations of SCs. This enables the proposed methods rotation invariant.

An agent processes an area of the image, its goal is to detect an object and check if this object belongs to the knowledge database.

To achieve the above goal, we have to build this architecture. Regarding the architecture, we have many choices to implement according to the already presented approaches in this field. Some of these approaches depend on the extraction of data from the image in the form of vectors, vectors indicating positions and vectors emoting intensity gradients for those positions, in order to compare in a later stage with the images found in the database. Another approach is related to neural networks where the architecture is designed of many layers that consist of neurons to compare to the saved images, and this approach will be used in our implementation Previous Work

The two variables, the transformation and the correspondence, in point matching problem are closely related. Once one variable is known, solution for the other is mostly straight forward. Consequently, if either variable can be independently determined, the matching problem can be

considered solved. There are methods in the literature that are designed to take this kind of independent approach — namely, solve for one of the variables alone without even introducing the other. From our perspective, the intimate prior relationship between the two variables has been used, resulting in one variable dropping out of the formulation.

Other methods utilize this close relationship in a different way by treating point matching as a joint problem with both variables. With both variables in the picture, a simple alternating update process can then be implemented in the following manner: in phase one, one variable is held fixed and the other estimated and in phase two, the preceding sequence is reversed. During the alternation, each variable improved the other. The resulting algorithm is simple and usually yields good results provided the algorithm converge quickly.

Thus all point matching algorithms can be characterized by examining the way they handle the two variables. Insofar as a method attempts to solve either the correspondence or the transformation alone, it can be regarded as a independent estimation approach. On the contrary, if a method tries to find the solution for both variables, usually via an alternating update scheme, it can be regarded as a joint estimation approach. We use this distinction to orient our description of previous research.

Typically, these methods are designed for the point matching problem when only rigid transformations are involved. A search in the parameter space of rigid mappings is considered feasible due to the low dimensionality of the mapping. In 2D, a rigid/affine transformation has six parameters; and in the case of 3D, nine parameters are involved. Moment-of-inertia analysis [1] is a classical technique that attempts to solve for the transformation without introducing the notion of correspondence. Originated in physics, the idea is to find the center of mass and the principal axis of the data and use them to align the point-sets. The center of mass provides us with the information about the global location of the data and the principal axis with the information of global orientation. While

the former can be used to estimate the translation component of the transformation, the later aids in the estimation of the rotation. This method is simple and but usually only provides a rough alignment

A more sophisticated technique is the Hugh Transform [2,3]. The transformation parameter space is divided into small bins, where each bin represent a certain configuration of transformation parameters. The points are allowed to vote for all the bins and the bin which get the most votes is chosen. The answer typically rejects the “tyranny of the majority.” The voting procedure tolerates a reasonable amount of noise and outliers. There are numerous other methods such as tree searches [4,8 ], the Hausdorff Distance[5], Geometric Hashing [12,21] and the alignment method [6,7] as well.

All these methods work well only for rigid transformations. When it comes to non-rigid mappings, the huge number of the transformation parameters (often proportional to the size of the dataset) usually renders these methods ineffective. We suspect that this is the main reason for the relative dearth of literature in non-rigid point matching despite the long history of the problem for rigid, affine and projective transformations [8-14].

The second type of methods work with more sparsely distributed point-sets. Shape attributes based on either local or global context are defined in these methods. The attributes are then used to determine the correspondence.

The modal matching approach in [15] uses a mass and stiffness matrix that is built up from the Gaussian of the distances between any point feature in one set and the other. The eigenvectors of such matrices are ordered according their eigenvalues and are called mode shape vectors. The correspondence is computed by comparing each point’s relative participation in the Eigen-modes. The basic idea is that while a point-set of a certain shape is non-rigidly deforming, different points at different locations should have systematically different ways of movement. Such differences are used to distinguish the points and determine their correspondences. Robust point matching (RPM) was introduced by Gold et al. [10] the method performs registration using deterministic annealing and soft assignment of correspondences between point sets. Whereas in ICP the correspondence generated by the nearest-neighbor heuristic is binary, RPM uses a soft correspondence where the correspondence between any two points can be anywhere from 0 to 1, although it ultimately converges to either 0 or 1. The correspondences found in RPM is always one-to-one, which is not always the case in ICP.

**Methodology**

To overcome this accidental alignment problem, we propose a Feature detection & Matching approach called Putative Feature Point Matching (PFPM) algorithm consisting of the following the key ingredients:

- a) We detect salient contours using bottom-up segmentation or contour grouping. Long contours are more distinctive, and maintaining contours as integral tokens for matching removes many false positives due to accidental alignment.
- b) We break the model Feature into its informative semantic parts, and explicitly check which subset of model Feature parts is matched. Missing critical model parts can signal an accidental alignment between the image and model.
- c) We seek holistic Feature matching. We measure Feature features from a large spatial extent, as well as long-range contextual relationships among object parts. Accidental alignments of holistic Feature descriptors between image and model are unlikely. Our Putative Feature Point Matching

algorithm reduces to finding a maximal, holistic matching between a set of image contours and a set of model parts. It searches over figure/ground labelling of the image and model contours, and correspondences between them. It is important to note that, in general, image contours and model contours do not correspond one-to-one. The holistic matching occurs only by considering a set of ‘figure’ contours together. To formulate this set-to-set matching task, we define control points sampled on and around the image and model contours.

Following figure shows the system block diagram.

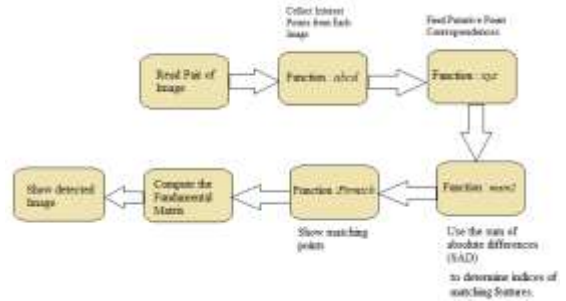


Figure 1 Block Diagram of System  
Flow chart of the system is shown in figure 2.

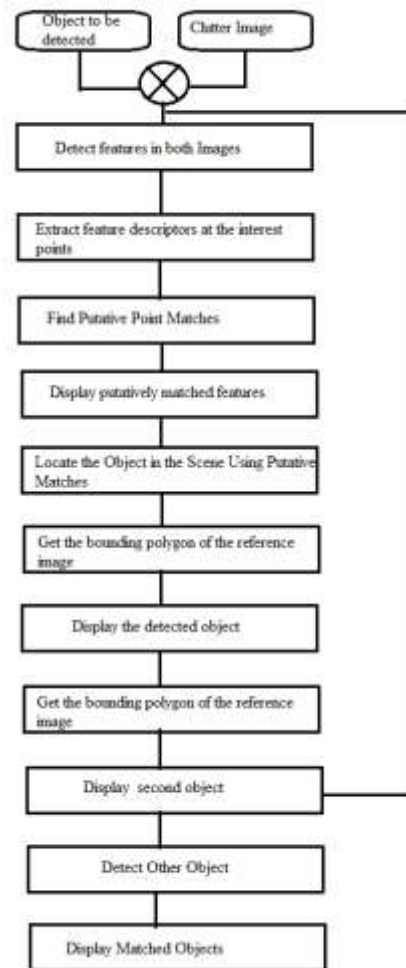


Figure 2. System Flowchart

**PFPM Algorithm**

- Read the reference image & target image.
- Detect Feature Points in both Images, Visualize the strongest feature points found in the target image.(Go to Step 1 of abcd )
- Extract feature descriptors at the interest points in both images. (Go to Step 1 of xyz)
- Find Putative Point Matches, Match the features using their descriptors. (Go to Step 1 of Ftrmtch & fundamentalff )

- Display putatively matched features.
- Locate the Object in the Scene Using Putative Matches Display the matching point pairs.
- Get the bounding polygon of the reference image.
- Display the detected object.
- Get the bounding polygon of the reference image Detect a second object by using the same steps as before.
- Repeat Steps 1-6
- Display Both Objects

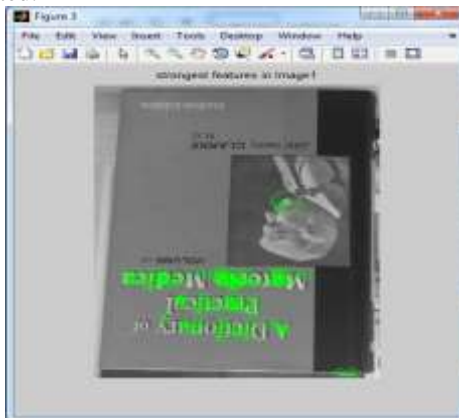
### Experimental Results

The output of each step after its execution is shown using screen shots of each stage. In Our Project, our main motive is to detect the object which has some degree of rotation. Means not exactly same in clutter image. In this example we rotate the image of book . our task is to detect the book in clutter image.



**Figure 3 Figure 2. Read the reference image & target image.**

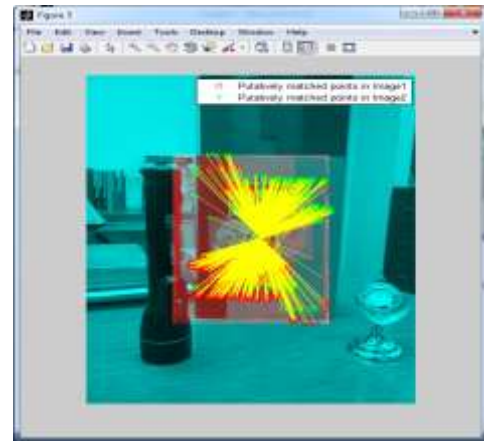
Detect Feature Points in both Images, Visualize the strongest feature points found in the target image. Here we Find Putative Point Matches, Match the features using their descriptors in both of Images i.e. Form Object to be detected & Cluttered Image. After finding putatively matched points we Display putatively matched features. We mark the strongly selected feature point using polygons. Here we define the number of strongest feature to be selected.



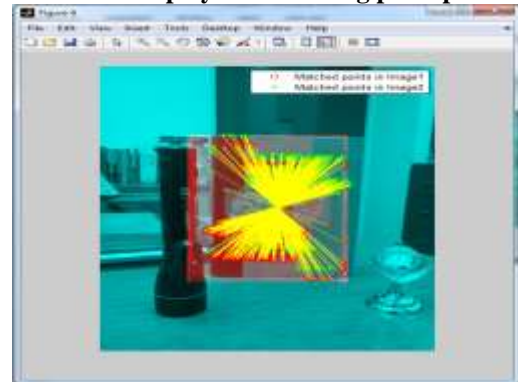
**Figure 4 strongest feature to be selected in object Image which is rotated by 180°**



**Figure 5 Figure 4. Strongest feature to be selected in reference image.**



**Figure 6 Locate the Object in the Scene Using Putative Matches Display the matching point pairs.**



**Figure 7 Matched Objects.**

### Conclusion

Here we presents an algorithm for detecting a specific object based on finding point correspondences between the reference and the target image. It can detect objects despite a scale change or in-plane rotation. It is also robust to small amount of out-of-plane rotation and occlusion.. Feature-based methods for Image registration frequently encounter the correspondence problem. Regardless of whether points, lines, curves or surface parameterizations are used, feature-based Image matching requires us to automatically solve for correspondences between two sets of features This method of object detection works best for objects that exhibit non-repeating texture patterns, which give rise to unique feature matches. This technique is not likely to work well for uniformly-colour objects, or for objects containing repeating patterns.

### Reference

- [1] N. Aronszajn. Theory of Reproducing Kernels. Trans. of the American Mathematical Society, 68(3):337–404, 1950.
- [2] A. Basu, I. R. Harris, N. L. Hjort, and M. C. Jones. Robust and Efficient Estimation by Minimising A Density Power Divergence. Biometrika, 85(3):549–559, 1998.
- [3] S. Belongie, J. Malik, and J. Puzicha. Shape Matching and Object Recognition Using Shape Contexts. TPAMI, 24(24):509–522, 2002.
- [4] P. J. Besl and N. D. McKay. A Method for Registration of 3-D Shapes. TPAMI, 14(2):239–256, 1992.
- [5] L. G. Brown. A Survey of Image Registration Techniques. ACM Computing Surveys, 24(4):325–376, 1992.
- [6] H. Chui and A. Rangarajan. A new point matching algorithm for non-rigid registration. CVIU, 89:114–141, 2003.
- [7] A. W. Fitzgibbon. Robust Registration of 2D and 3D Point Sets. Image and Vision Computing, 21:1145–1153, 2003.
- [8] P. J. Huber. Robust Statistics. John Wiley & Sons, New York, 1981.

- [9] B. Jian and B. C. Vemuri. Robust Point Set Registration Using Gaussian Mixture Models. *TPAMI*, 33(8):1633–1645, 2011.
- [10] A. E. Johnson and M. Hebert. Using Spin-Images for Efficient Object Recognition in Cluttered 3-D Scenes. *TPAMI*, 21(5):433–449, 1999.
- [11] J.-H. Lee and C.-H. Won. Topology Preserving Relaxation Labeling for Nonrigid Point Matching. *TPAMI*, 33(2):427–432, 2011.
- [12] X. Li and Z. Hu. Rejecting Mismatches by Correspondence Function. *IJCV*, 89(1):1–17, 2010.
- [13] D. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *IJCV*, 60(2):91–110, 2004.
- [14] B. Luo and E. R. Hancock. Structural Graph Matching Using the EM Algorithm and Singular Value Decomposition. *TPAMI*, 23(10):1120–1136, 2001.
- [15] J. Ma, J. Zhao, Y. Zhou, and J. Tian. Mismatch Removal via Coherent Spatial Mapping. In *ICIP*, 2012.