Keypatches: A New Type of Local Features for Image Matching and Retrieval

Andrzej Śluzek
Nanyang Technological University
School of Computer Engineering
Blk N4 Nanyang Avenue
Singapore 639798

assluzek@ntu.edu.sg

ABSTRACT

The paper presents a new approach in defining (and detecting, as an additional option) local features that can be used for matching images and/or visual information retrieval. The method is based on the moment-based pattern detectors presented in previous papers. The proposed local features (preliminarily called *keypatches*) are obtained by approximating circular windows located at keypoints pre-detected using any typical detector with a selection of geometric patterns. At each keypoint, the optimum approximations (for all available patterns) of the window are computed using moment-based equations. For any approximation, its similarity to the actual window content is estimated. The *keypatch* is defined if a sufficiently accurate approximation exists. Keypoints where the window cannot be approximated with the sufficient accuracy are ignored. If no pre-detector of keypoints is available, the method itself can find the initial locations of the keypoints. The proposed approach is suitable for both grey-level and colour images (though the latter are only briefly discussed in the paper). Exemplary results explaining the method and illustrating its performances are included and discussed.

Keywords

Local features; keypoints; geometric patterns; moments; Radon transform; image similarity.

1. INTRODUCTION

Visual information retrieval (VIR) and vision-guided search/detection in natural environments are challenging problems but the areas of their applications are rapidly growing. Although the physical constraints of those applications might be completely different (e.g. online face identification in a crowded place *versus* search in a digitized database of press video-clips) from the machine vision perspective the underlying problems are rather similar. In general, the goal is to extract a rather small fragment of visual data from a large amount of background, and to match the fragment to a relevant piece of available visual information. Usually, both the visual query (i.e. a photo of the wanted person, pictures of the object of interest, etc.) and the available visual data (camera-captured images, database images, etc.) are significantly distorted and/or corrupted. Partial occlusions, poor and/or diversified illumination, scale changes, perspective distortions, etc. are the most typical examples of conditions deteriorating the informational quality of visual data.

There is a general consent that under such conditions visual data should be processed and analyzed using the local approach. This belief is strongly supported by the evidence (e.g. [Bie87a] and [Ede97a]) that humans recognise known objects as groups of local visual clues which, if enough of them can be found, are "interpolated" into the object. Although this scheme may not work perfectly (e.g. well-known optical illusions) but generally it allows detection of known objects under the degrading conditions listed above. The machine vision systems often try to emulate this approach.

Local features (also known as corner points, keypoints, interest points, characteristic points, local visual saliencies, etc.) have been used in computer vision since the 80's (e.g. [Har88a]). Their applications have been gradually expanded from stereovision and motion tracking to image matching and detection of known objects in difficult conditions, e.g. [Sch97a]. Once the weak points of early keypoint detectors (e.g. scale changes and perspective distortions) have been resolved, they became a relatively powerful tool for VIR and vision-guided navigation. Currently, SIFT (see [Low04a]) is considered the most successful detector though others (e.g. modifications of Harris-Plessey or SUSAN, see [Smi92a]) are also popular.

Typical keypoint detectors are generally based on differential (or quasi-differential, e.g. SUSAN) properties of the intensity (or colour) functions. Even though keypoints can be further characterized by various descriptors (e.g. [Low04a] or [Isl05a]) those characteristics do not contain any information about the (visual) semantics of the keypoints. Thus, whenever keypoints are matched against database keypoints, a full search should be initially attempted. Another practical disadvantage of typical detectors is a large number of keypoints extracted even from images of moderate complexity. Although this may improve chances of finding matching keypoints in the correspondingly similar images, but hundreds or thousands of keypoints to be matched against similar numbers of keypoints in each database image require huge computational resources.

In this paper, we discuss a method that incorporates visual semantics into the characteristics of keypoints. We assume that a patch (we use only circular or square patches) around a keypoint may be similar to a certain geometric pattern. The optimum approximations of the patch by given patterns are found (using moment-based expressions, see the explanations in Section 2). The list of patterns used for the approximations can be arbitrarily reduced or expanded. Subsequently, the level of (visual) similarity between the patch and its pattern-based approximations is determined using the Radon transform, as explained in Section 3. The *keypatch* is created if the patch around the keypoint is sufficiently accurately approximated by at least one pattern. If the patch cannot be accurately approximated by any pattern, no keypatch is produced. The geometric and colour/intensity characteristics of those accurate approximations are very powerful local descriptors of the image content. Explanations on how keypatches can be effectively used for a quick matching of even very complex images are given in Section 4.

If no pre-detector of keypoints is available, the method can produce the initial list of keypoints using its own algorithm. This is briefly explained in Section 3 as more details are available in our previous papers.

2. PATTERNS & APPROXIMATIONS

Previously published papers (e.g. [Slu05a], [Slu07a]) described a method that can build the optimum (in a certain sense) approximation for a circular (grey-level or colour) image of radius $\boldsymbol{R}$ by predefined patterns of the same radius. Such patterns could be corners and corner-like structures (e.g. junctions) but the method is generally applicable to any parameter-defined patterns. Fig. 1 presents several exemplary circular patterns of radius $\boldsymbol{R}$.

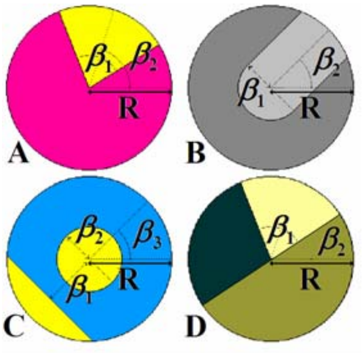

Figure 1. Exemplary circular patterns.

Each instance of such a pattern is characterised by several geometric (configuration) parameters (represented by indexed β's in Fig. 1) and several colours/intensities. For example, a T-junction (see Fig.1D) can be uniquely defined by three colours/intensities, the orientation angle β_2 and the angular width β_1.

More complex patterns can be proposed as well, but within this paper we discuss only rather simple patterns with 2-3 configuration parameters (and a similar number of colours/intensities) as such pattern combine low complexity with sufficiently useful structural information.

For such circular patterns (and for any circular images) various functions can be calculated over the whole area of the circle. The formulae for those functions are defined by the structure of a pattern, while the configuration and colour/intensity parameters appear as variables within the formulae. We propose to use ordinary moments (because of low complexity, low sensitivity to noise, etc.) as the functions of interest. Thus, moments of various orders are calculated for the circular images (with the assumed origin of coordinates in the circle's centre). Examples of moment expressions for selected patterns can be found in [Slu05a].

The moment expressions, when compared to the actual moment values for a given instance of the pattern form a system of equations with the pattern parameters as the unknowns. Therefore, given a sufficient number of equations, the values of the parameters can be retrieved as solutions of the system. More details are given in the references ([Slu05a] and other papers) but the following examples are given as illustrations. They give solutions for selected configuration parameters of **A**, **B**, **C** and **D** patterns shown in Fig. 1. Note that some solutions are given for colour images for which moments are 3D vectors of in RGB space, and some for grey-level images where moments are scalars:

$$\beta_1(A) = 2\arcsin\sqrt{1-\frac{16\left(\left\|\overrightarrow{m_{20}}-\overrightarrow{m_{02}}\right\|^2+4\left\|m_{11}\right\|^2\right)}{9R^2\left(\left\|m_{10}\right\|^2+\left\|m_{01}\right\|^2\right)}} \quad (1)$$

$$\beta_2(B) = \operatorname{atan2}(\pm m_{01}, \pm m_{10}) \quad (2)$$

$$\beta_1(C) = \sqrt{\frac{\left(\overrightarrow{m_{20}}-\overrightarrow{m_{02}}\right)^2+4\left\|\overrightarrow{m_{11}}\right\|^2}{\left(\left\|\overrightarrow{m_{10}}\right\|^2+\left\|\overrightarrow{m_{01}}\right\|^2\right)}} \quad (3)$$

$$m_{01}\cos\beta_2(D) - m_{10}\sin\beta_2(D) = \pm\frac{4}{3R}\sqrt{(m_{20}-m_{02})^2+4m_{11}{}^2} \quad (4)$$

Intensity/colour parameters of the pattern instances can be found using other moment-based formulae (also individually designed for each pattern). Examples and explanations can be found in [Slu05a], [Slu07a].

Generally, the moments of only low orders should be used (higher-order moments are too sensitive to noise) so that only up to 6 equations can be formed from moments of order 0, 1 and 2 so that only patterns with just a few parameters can be handled. In order to increase the number of equations (i.e. to allow the usage of complex patterns with more parameters) we propose to calculate moments of the circle's quarter and halves (as shown in Fig. 2).

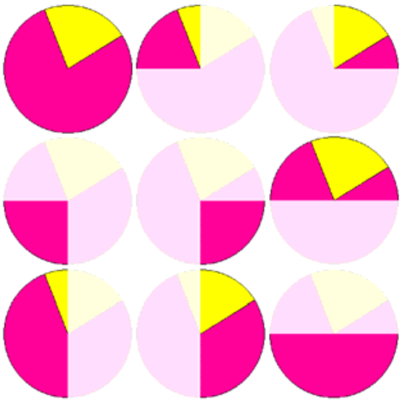

Figure 2.

Thus, for any circular image 54 moments of order 0, 1 and 2 are computed (six moments of the whole circle, six moments of the upper half, six moments of the first quarter, etc.). Then, systems of equations with many unknowns can be formed and solved (i.e. patterns with many parameters can be used) using only the low-order moments. Additionally, simple symmetrical patterns for which many moments disappear can be handled as well. In the presented examples, however, there is no need to use moments of the halves or quarters.

2.1. Pattern-based Approximations

It can be noted that the solutions for pattern parameters (exemplified by Eqs 1-4) actually may exist not only for the circular images containing the pattern of interest but to any circular image as well. If such solutions are found for an arbitrary circular image, they define the *optimum approximation* of the image by the given pattern. If the solution does not exist (e.g. the square root of a negative number in Eq.1) or the solution is physically impossible the circular image cannot be approximated by a given pattern.

Several examples of diversified circular images and their approximations by various patterns are given in Fig. 3.

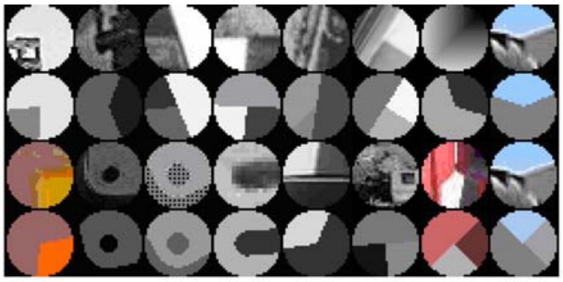

Figure 3. Various circular images (rows 1 and 3) and their approximations (rows 2 and 4, respectively) by selected patterns.

Fig. 3 shows that for images containing actual patterns the corresponding approximations obviously look very similar. However, there are also examples of approximations looking very different than the images. In some cases, the results are ambiguous. If the image content is inconclusive it may me approximated by various patterns with visually similar accuracy (e.g. the last column in Fig. 3).

3. KEYPATCHES

3.1. Similarity Measure

Examples given in Fig. 3 indicate that pattern-based approximations of circular images vary in their accuracy (i.e. visual similarity to the original images). To objectively quantify the accuracy of such approximations, a measure is needed to compare a circular image and its pattern-based approximation. Various measures have been proposed in out previous paper (e.g. using directly the intensity/colour differences in [Slu05a] or using moment similarities in [Slu07b]). It was eventually found that the most accurate results can be obtained using a well-known Radon transform (e.g. [Dea83a]) calculated for several directions. The performed tests indicate that satisfactory results can be achieved using the Radon transform for just two directions: horizontal (along OX axis) and vertical (along OY axis).

General expressions for the horizontal and vertical Radon transforms can be derived analytically for each pattern so that the transformations is calculated only for the processed circular images. Fig. 4 compares the horizontal and vertical Radon transforms for an accurate corner approximation (Fig. 4A), a poor approximation (Fig. 4B) and for an approximation of average quality (Fig. 4C).

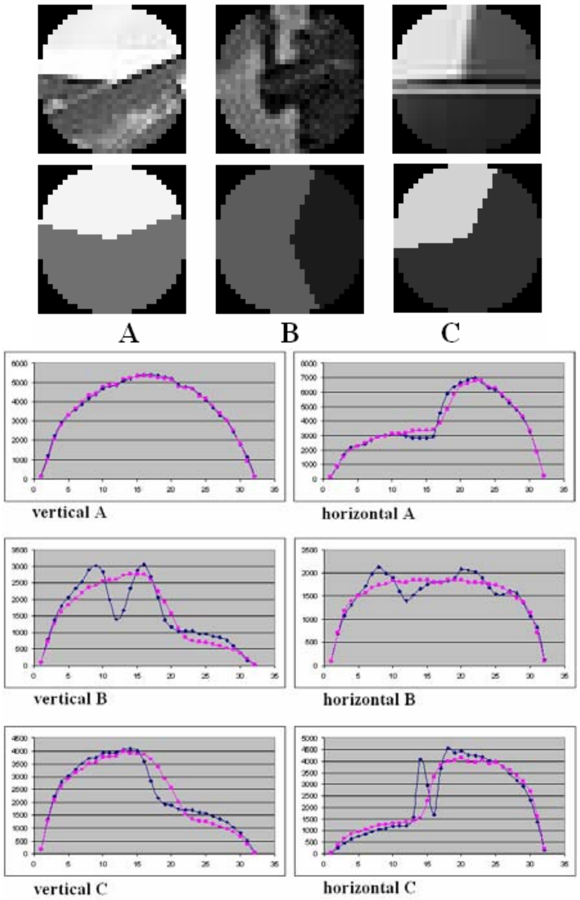

Figure 4. Vertical and horizontal Radon transforms for three images and their corner approximations of good (A), poor (B) and average (C) quality.

The difference between ***I*** circular image and its ***App*** approximation (both of radius ***R***) can be expressed as

$$\begin{aligned} diff(I,App) = \int_{-R}^{R} & \left|Rad_I(0^\circ,s) - Rad_{App}(0^\circ,s)\right| + \\ & + \left|Rad_I(90^\circ,s) - Rad_{App}(90^\circ,s)\right| ds \end{aligned} \qquad (5)$$

where ***Rad*** indicates the Radon transform.

Actually, the difference is normalized by the image's 0th moment $\boldsymbol{m_{00}(I)}$ so that the difference does not increase for light images. The difference is also further normalized using the contrast between approximation intensities/colours. Otherwise, poorly contrasted images would be more similar to the approximations than high-contrast images.

3.2. Defining Keypatches

Keypatches are searched for using in image with a number of pre-detected keypoints (the variation where the proposed method by itself detects both keypoints and keypatches will be briefly discussed later). At each keypoint a circular window is located which is subsequently approximated by the available patterns. Fig. 5 shows two exemplary images with their keypoints detected using the improved Harris detector, [Sch00a].

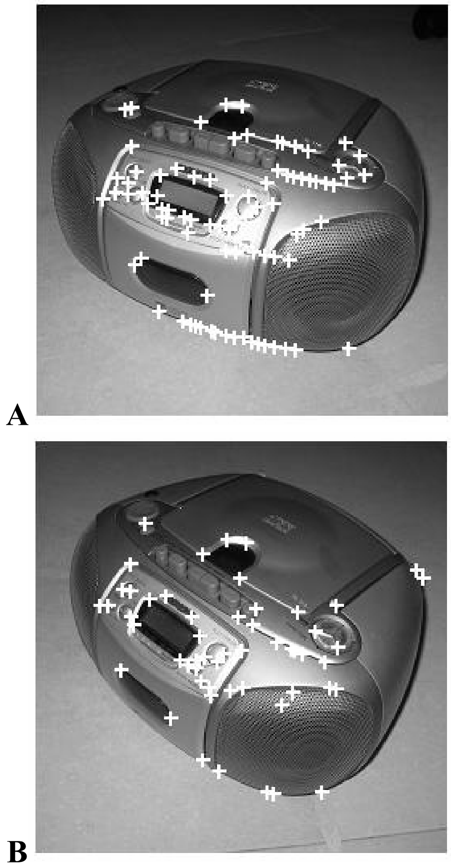

Figure 5. Two images with keypoints detected by the improved Harris detector.

The radius for the circular windows is determined by several factors. Generally, the selected radius depends on how the keypoints are pre-detected and what is the expected content of images. For example:

(a) Scale-invariant keypoint detectors (e.g. SIFT, [Low04a]) return the optimum scale for each keypoint. In such cases, the window radius ***R*** should be proportional to this scale (i.e. the radius is generally different for each keypoint).

(b) When the scales are not available, the radius should depend of the type of analyzed images. If images potentially contain general views of the observed scenes, the recommended radius is rather small (so that each window contains only fine details and not too much of the background). However, if close-ups of the objects are expected within images, longer radii are recommended because important details might be too large to fit into small windows.

Keypatches are defined by (*keypoint*, *pattern*) pairs. A keypoint ***K*** becomes a keypatch for the pattern ***Pt*** if the following conditions are satisfied:

1. Approximation ***App***(***Pt***, ***K***) exists, i.e. the circular window ***I*** located at ***K*** keypoint can be approximated by ***Pt*** pattern.
2. The difference ***diff***(***I***, ***App***(***Pt***,***K***)) – see Eq.5 –is smaller than any other difference between ***I*** and an approximation by another pattern.
3. The difference ***diff***(***I***, ***App***(***Pt***,***K***)) is smaller the a predefined threshold.

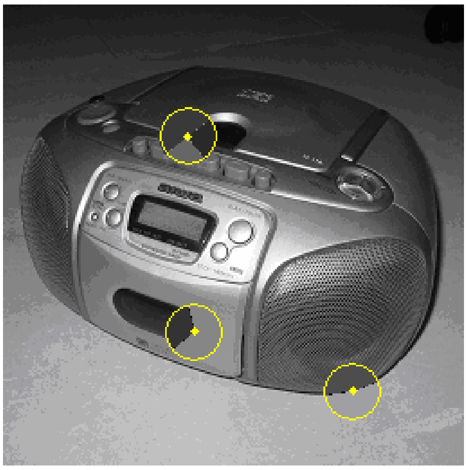

Figure 6. Three exemplary keypatches found for the image from Fig. 5A.

It is obvious, therefore, that not all keypoints would eventually produce keypatches. For some keypoints the approximations may not exist (i.e. equations like Eqs 1-4 or similar have no solutions) while for other keypoints none of the approximations is sufficiently accurate. One keypoint may eventually produce two (or even more) keypatches if approximations for several different patterns are sufficiently similar to the circular window content.

Fig. 6 shows three keypatches found in the upper image from Fig. 5 (note that the corresponding image fragments are replaced by the approximations). It should be noted that the keypatches do not approximate the image fragments too accurately. However, within the set of available patterns (actually only four patterns have been used in this experiment) they are the best approximation of the selected keypoints. Moreover, they also satisfy the above Condition 3 since the threshold used is not very restrictive.

3.2.1 Keypatches in ColourImages

The procedure of detecting keypatches in colour images is generally the same. The only difference is that three colour components (we have used RGB but other colour spaces are also acceptable) are used to calculate the configuration parameters (see Eqs 1-4) and the optimum colours of the approximations. It actually improves the method’s flexibility as the moments involved are separately computed for each colour components. More discussion can be found in [Slu07a].

The difference between circular images and their approximations is determined by the Radon transform based on colour vectors (rather than intensity scalars). Therefore, in Eq.5 the absolute values should be replaced by the vector norms. Otherwise, the colour images are analysed identically.

3.2.2 Self-detection of Keypoints

In the previous sections we have assumed that the keypoints used to define and detect keypatches are pre-detected by a certain keypoint detector. Actually, the method can self-detect the keypoints using the same concept of pattern-based approximation. The principles of keypoint self-detection are as follows:

The whole analysed image is scanned using a circular of the selected radius (see explanations in the initial part of Subsection 3.3). For each location of the window, the pattern-based approximations are built for the available pattern and the difference between the approximations and the window content is estimated. A keypoint is places at all locations where the difference between the window content and a pattern-based approximation for a certain pattern reaches a local minimum. Therefore, the keypoints are immediately associated with the type of pattern and the subsequent detection of keypatches is predefined by this association.

Although this approach seams computationally complex, actually the moment calculations (the most intensive calculations in the algorithm) are highly reusable when the circular window scans an image. Eventually, the overall complexity can be compared to other keypoint detectors (e.g. SIFT).

The algorithm of keypoint self-detection is not further discussed in this paper but more explanations can be found is our previous publications (e.g. [Slu07a], [Slu07b]). It should be highlighted, however, that by using this approach keypoints are not located as accurately as by differential keypoint detectors. This is because moments are integrals. When the image is scanned the approximations (and their similarities to the image content) change smoothly. Therefore, noise effects further away from the window centre could disturb the results so that the minimum difference between the circular window and its approximation may be found not exactly at the perceived location of the keypoint. As an illustration, Fig. 7 shows how blurred the difference *diff* (see Eq.5) values are even for a perfect image (containing corners) where the corner keypoints can be located very accurately using alternative detectors.

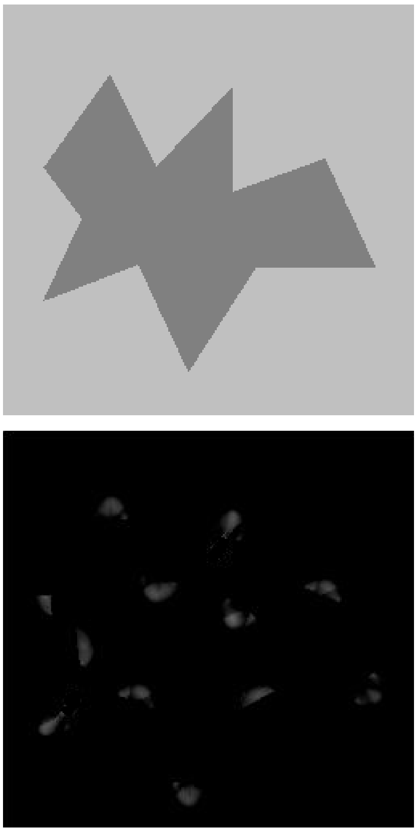

Figure 7. The changes of *diff* function for a perfect image approximated by the *corner* pattern.

However, this self-detection of keypoints can be recommended for blurred images of poor quality where derivative-based keypoint detectors may not work reliably enough.

4. DISCUSSION

4.1. Using Keypatches for Image Matching

The main advantage of keypatches is that they add visual semantics to the concept of keypoints. “Traditional” keypoints are locations where the intensity/colour functions exhibit certain differential properties but no semantically meaningful visual interpretation can be added to such keypoints.

From the perspective of image matching the following advantages of keypatches should be highlighted:

(a) Matches for a given keypatch from a query image should be searched for only within the database keypatches of the same pattern. Even if the overall number of keypatches as large as the number of keypoints, they are categorized based on the pattern type and matching is always limited to the same category. Thus, the estimated number of attempted matches is reduced exponentially.

(b) Database images can be annotated using keypatches (usually the most accurate and/or the most prominent). Thus, keypatches define local geometric structures present in images.

(c) Description/annotation of already processed database images can be easily enhanced when new patterns become available and thus keypatches of new types can be detected.

(d) Image search can be done more selectively and/or adaptively by using only selected categories of keypatches. For example, if a query image contains keypatches of certain categories, the database search would be limited to the keypatches of those categories only.

(e) Intensity/colour and geometric characteristics of keypatches can be used to effectively extract groups of consistently matched keypatches. Such consistent groups of keypatches presumably represent the same object present in both images, even if the object is partially occluded and photometrically distorted. The actual significance of intensity/colour and configuration parameters can be adaptively changed for various application.

As an illustrative example, selected locations from Figs 5A and 5B are matched in Fig. 8. For all matched location keypatches have been found (some of them are shown in Fig. 6) and the matching process has been based on the keypatches.

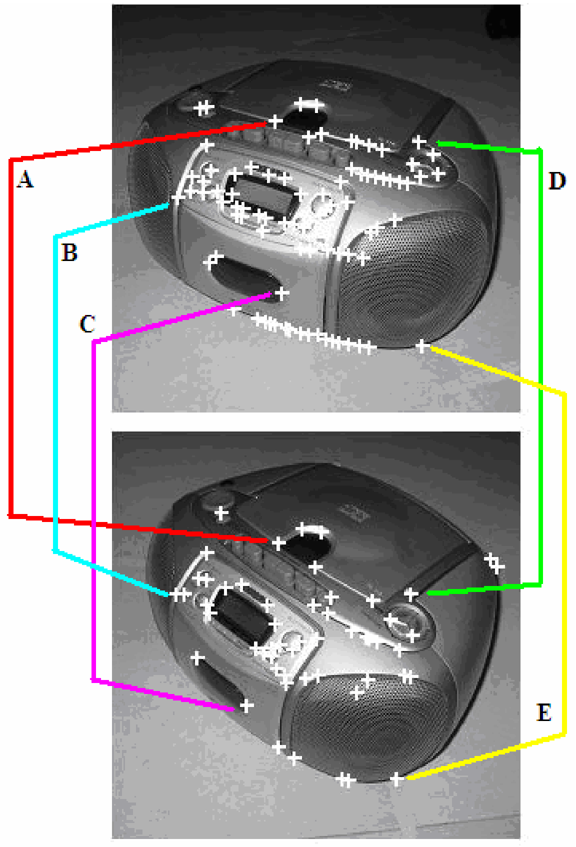

Figure 8. Selected matches for Fig. 5 images.

Table 1 compares the characteristics of the matched pairs. Although details of the image matching techniques are not discuss in this paper (more information is available in [Isl05a] and other publications) the content of the table – in particular the consistency between orientational parameters of individual keypatches and the whole group – is a strong indicator that all keypatches may belong to the same object. Additionally, the relative transformation of the object between both images can be roughly estimated from the spatial distribution of the keypatch locations and their configuration parameters.

It should be noted that the high level of consistency seen in Table 1 has been achieved only for the patterns associated with the keypatches. For other patterns (actually only four types of patterns have been used) the approximations of for the selected keypoints also exist (in most cases). However, the accuracy of those approximations was poorer and, secondly, the approximations obtained for the matched pairs were significantly different.

A meaningful and semantically correct matching process for complex images containing multiple objects is, in general, not straightforward. Although the researches in this area have only started, we envisage a significant usage of artificial intelligence techniques (e.g. neural networks and approximate reasoning) and feedback loops between low- and high-level analysis.

Pair A	Pattern: **90° T-junction**		

lower	Intensities:
	A = 66
	B = 98
	C = 33
		Orientation:
		β = 65°	

upper	Intensities:
	A = 55
	B = 100
	C = 41
		Orientation:
		β = 59°	

Pair B	Pattern: **Angle**		

lower	Intensities:
	A = 127
	B = 69
		Orientation:
		β = 3°
			Angular width:
			α = 141°

upper	Intensities:
	A = 143
	B = 69
		Orientation:
		β = -5°
			Angular width:
			α = 150°

Pair C	Pattern: **X-cross junction**		

lower	Intensities:
	A = 92
	B = 129
		Orientation:
		β = 81°	

upper	Intensities:
	A = 88
	B = 127
		Orientation:
		β = 68°	

Pair D	Pattern: **Angle**		

lower	Intensities:
	A = 115
	B = 70
		Orientation:
		β = 92°
			Angular width:
			α = 81°

upper	Intensities:
	A = 95
	B = 56
		Orientation:
		β = 84°
			Angular width:
			α = 88°

Pair E	Pattern: **Angle**		

lower	Intensities:
	A = 78
	B = 168
		Orientation:
		β = -103°
			Angular width:
			α = 174°

upper	Intensities:
	A = 77
	B = 162
		Orientation:
		β = -109°
			Angular width:
			α = 173°

Table 1. Comparison of keypatch parameters for the match image locations shown in Fig. 8.

4.2. Concluding Remarks

The paper presents a sketchy description of a novel technique of defining local features. The features, so-called *keypatches*, are defined over a set of keypoints pre-detected by other keypoint detectors. Although the proposed technique can self-detect keypoints as well, this approach is only marginally mentioned in the paper.

A keypatch indicates the location where locally the image content can be satisfactorily accurately approximated by one of available geometric patterns (including full geometric and colour/intensity specification of that approximating pattern). Thus, a semantically meaningful local specification of the image content (image annotation) is possible.

The method could be particularly useful for VIR (visual information retrieval) where large collections of complex databases are searched for images containing fragments similar to specified or unspecified areas of query images. The second intended application is autonomous navigation where agents exploring unknown environments should locate known objects and/or should characterize the observed scene in terms of its similarity to already known environments.

The paper focuses on the fundamentals of the method, i.e. the problem formulation, basic definitions and the mathematical principles of defining and detecting keypatches. Examples are included to explain and illustrate those fundamentals.

Currently, the method is developed into a working platform that can be used for selected applications. One of the important issues is expansion of the list of available pattern so that complex images can be annotated with larger numbers of diversified keypatches. The results of currently conducting researches, regarding both the theoretical background and experimental applications, will be addressed in future papers.

5. REFERENCES

[Bie87a] Biederman, I., Recognition-by-components: A theory of human image understanding, *Psychological Review*, Vol. 94(2), pp. 115-147, 1987.

[Dea83a] Deans, S.R., *The Radon Transform and Some of Its Applications*. John Wiley & Sons, New York, 1983.

[Ede97a] Edelman, S., Computational theories of object recognition, *Trends in Cognitive Sciences*, Vol. 1(8), pp. 298-309, 1997.

[Har88a] Harris, C. and Stephens, M., A combined corner and edge detector, *Proc. of 4th Alvey Vision Conference*, pp 147-151, 1988.

[Isl05a] Islam, M.S., Sluzek, A., Zhu, L., Detecting and matching interest points in relative scale. *Machine Graphics & Vision*, vol. 14(3), pp. 259-283, 2005.

[Low04a] Lowe, D., Distinctive image features from scale-invariant keypoints, *Int. Journal of Computer Vision*, Vol. 60(2), pp 91-110, 2004.

[Sch97a] Schmid, C. and Mohr, R., Local grayvalue invariants for image retrieval, *IEEE Trans. PAMI***,** Vol. 24(5), pp. 530-535, 1997.

[Sch00a] Schmid, C., Mohr, R. and Bauckhage, C., Evaluation of interest point detectors. *Int. Journal of Computer Vision*, Vol. 37(2), pp 151-172, 2000.

[Slu05a] Sluzek, A., On moment-based local operators for detecting image patterns, *Image and Vision Computing*, Vol. 23(3), pp 287-298, 2005.

[Slu07a] Sluzek, A. Approximation-based keypoints in colour images: A tool for building and searching visual databases", *LNCS*, Vol.4781 (eds G Qiu et al), Springer Verlag, 2007 (in print).

[Slu07b] Sluzek, A., Islam, M.S. New types of keypoints for detecting known objects is visual search tasks. In: *Vision Systems, Application* (eds Obinata, G. and Dutta, A.), I-Tech, Vienna, pp 423-442, 2007.

[Smi97a] Smith, S.M. and Brady, M., SUSAN – a new approach to low level image processing. *Int. Journal of Computer Vision*, Vol. 23(1), pp. 45-78, 1997.