A Scale Invariant Keypoint Detector Based on Visual and Geometrical Cues

Levi O. Vasconcelos(✉), Erickson R. Nascimento, and Mario F.M. Campos

Department of Computer Science, Universidade Federal de Minas Gerais, Belo Horizonte, Brazil
{leviovasconcelos,erickson,mario}@dcc.ufmg.br

Abstract. One of the first steps in a myriad of Visual Recognition and Computer Vision algorithms is the detection of keypoints. Despite the large number of works proposing image keypoint detectors, only a few methodologies are able to efficiently use both visual and geometrical information. In this paper we introduce KVD (Keypoints from Visual and Depth Data), a novel keypoint detector which is scale invariant and combines intensity and geometrical data. We present results from several experiments that show high repeatability scores of our methodology for rotations, translations and scale changes and also presents robustness in the absence of either visual or geometric information.

Keywords: Keypoint detector · Local features

1 Introduction

Over the years, the task of selecting a set of points of interest in images has been omnipresent in a large number of Visual Recognition and Computer Vision methodologies. A careful choice of points in images may avoid the inclusion of noisy pixels and enables the identification of regions that are rich in information, aiding an effective description of such regions. Additionally, the use of an image subset enables the tackling of cluttered backgrounds and occlusions in object recognition [7,3] and scene understanding applications. Moreover, the ever growing volume of data, which includes high resolution images, RGB-D data and the massive image repositories available in the web, makes the development of keypoint detectors crucial for a large number of image processing techniques.

In a common image representation pipeline for matching and classification tasks, before computing feature vectors for pixels, these pixels must be selected by a detector algorithm. Thus, while a descriptor algorithm is concerned with providing a discriminative identification for a keypoint by analyzing its vicinity, a detector is designed for finding informative image patches.

As stated, the detection of a set of points of interest, henceforth referred to as *keypoints*, consists in looking for points located in discriminative regions of

This work is supported by grants from CNPq, CAPES and FAPEMIG.

A. Pardo and J. Kittler (Eds.): CIARP 2015, LNCS 9423, pp. 341–349, 2015.
DOI: 10.1007/978-3-319-25751-8_41

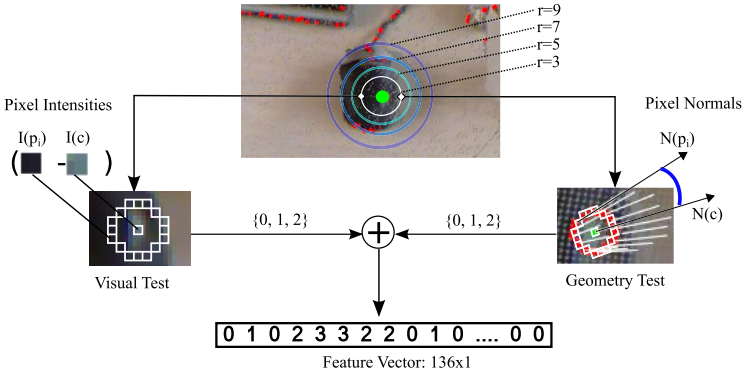


Fig. 1. The extraction and fusion of visual and geometrical features of KVD detector.

the image, that will account for good *repeatability*, which in turn may lead to smaller ambiguity. There is a vast body of literature on image keypoint detectors, of which [7,4,12,13] are well known representatives.

Broadly speaking, the main task of a keypoint detector is to assign a saliency score to each pixel of an image. This score is then used to select a (usually smaller) subset of pixels that presents the following properties: i) Repeatability; ii) Distinctiveness; iii) Locality and iv) Accurately localizable.

The main contribution of this paper is a scale invariant keypoint detector called KVD (Keypoints from Visual and Depth Data), which efficiently combines intensity and depth. Our method produces the best performing detector by combining visual and geometrical data, and presents a good performance and graceful degradation even in the absence of either one of them.

Related Work. Since the seminal paper of Morevec [10], where he presented one of the first corner detectors, a large number of keypoint detectors have been proposed. Harris detector [4], Harris-Laplacian [8], SIFT [7], SURF [1] are some of the most popular detectors for images.

A recent approach that has become popular in keypoint selection is based on machine learning techniques. Rosten and Drummond [12] proposed the FAST detector, which creates a feature vector that is used by a decision tree to classify the pixel as a keypoint. The Roten and Drummond's technique was improved by Rublee et. al [13]. They presented the ORB detector which uses a scale pyramid to add scale invariance and measures the cornerness of each keypoint candidate by computing Harris corner. Another recent methodology also based on machine learning technique is presented in [5]. The authors proposed a keypoint detection from depth maps by using Random Forest which is trained to maximize the repeatability score.

Extracting data from images can usually provide rich information on the object features. The main drawback is the sensitiveness of these feature to illumination changes. Geometrical information produced by 3D sensors based on structured lighting or time of flight, in its turn, is less sensitive to visible

lighting conditions. Three-dimensional data has been successfully used by algorithms such as NARF [15] and on 3D detectors implementations derived from 2D approaches [14] such as SIFT3D, HARRIS3D and HARRIS6D.

Despite the growing popularity of techniques that combine both visual and geometric informations to build descriptors [11,17] and their use in recognition tasks, this fusion is not a common approach for keypoint detection. In this work, we present a novel keypoint detector, which simultaneously takes into account both visual and geometrical information to detect keypoints.

We show that using both visual and geometric information at the detection level improves the quality and performance of higher level visual processes.

2 Methodology

The input for our algorithm is a pair (I, D), which denotes the output of a typical RGB-D device. For each pixel $\mathbf{x}$, $I(\mathbf{x})$ is the pixel's intensity, $D(\mathbf{x})$ is depth for that pixel, $P(\mathbf{x})$ is the corresponding 3D point, and $N(\mathbf{x})$ is its normal vector.

Our technique is built upon a supervised training approach, with a training step where a decision tree is created to classify points into keypoints and non-keypoints. There are three steps: the feature vector extraction and fusion, the model training and the non-maximal suppression.

Feature Extraction. The first step of the detection process creates a feature vector for every keypoint candidate. Figure 1 depicts the feature vector construction. Given an image pixel coordinates $\mathbf{c} \in \mathbb{R}^2$, we consider its vicinity as the image patches that contain the circles centred at $\mathbf{c}$ with radii varying in $r \in \mathcal{S}$. Each circle is defined by the function $B(r, \mathbf{c})$ which we denote as $B_r(\mathbf{c})$:

$$B_r(\mathbf{c}) : \mathbb{R}^3 \rightarrow \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_n\}. \tag{1}$$

The $B_r(\mathbf{c})$ function outputs all pixels $\mathbf{p_i}$ lying in the Bresenham's circle with radius equals to r. Thus, the vicinity considered consists of the concatenation of all vectors $B_r(\mathbf{c}), \forall r \in \mathcal{S}$. We define the vicinity of a central pixel $\mathbf{c}$ as:

$$\mathcal{V}_c = \{B_{r_1}(\mathbf{c}), B_{r_2}(\mathbf{c}), \ldots, B_{r_{|\mathcal{S}|}}(\mathbf{c})\}, \quad \forall r_i \in \mathcal{S}. \tag{2}$$

Whereas in this work, we used $\mathcal{S} = \{3, 5, 7, 9\}$. Thus, we compute visual features using fast intensity difference tests [12]. For each pixel $p \in \mathcal{V}_c$ and a given threshold t_v we evaluate:

$$\tau_v(\mathbf{c}, \mathbf{p}_i) = \begin{cases} 2 & \text{if } I(\mathbf{p}_i) - I(\mathbf{c}) < -t_v \\ 1 & \text{if } I(\mathbf{p}_i) - I(\mathbf{c}) \geq t_v \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

We embed geometric cues into the feature vector computed by the function τ_v to increase robustness both to illumination changes and to the lack of texture in the scenes. The geometric feature extraction $\tau_g(.)$ function is based on two invariant geometric measurements: i) the normal displacement, and ii) the surface's

convexity. The normal displacement test evaluates if the dot product between the normals $N(\mathbf{c})$ and $N(\mathbf{x_i})$ is smaller than a given displacement threshold t_g, and the convexity test computes the local curvature indicator, κ as:

$$\kappa(\mathbf{c}, \mathbf{p}_i) = \langle P(\mathbf{c}) - P(\mathbf{p}_i), N(\mathbf{c}) - N(\mathbf{p}_i) \rangle, \tag{4}$$

where $\langle . \rangle$ is the dot product, and $P(\mathbf{c})$ is the $3D$ spatial point associated with pixel $\mathbf{c}$ and depth $D(\mathbf{c})$. The κ function captures the convexity of geometric features and also unambiguously characterizes the dot product between surface normals. Thus, the geometrical features are computed as:

$$\tau_g(\mathbf{c}, \mathbf{p}_i) = \begin{cases} 2 & \text{if } \langle N(\mathbf{p}_i), N(\mathbf{c}) \rangle < t_g \wedge \kappa(\mathbf{c}, \mathbf{p}_i) > 0 \\ 1 & \text{if } \langle N(\mathbf{p}_i), N(\mathbf{c}) \rangle < t_g \wedge \kappa(\mathbf{c}, \mathbf{p}_i) < 0 \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Scale invariance is endowed to our detector by using the geometry information available in the depth map to weigh the influence of each circle. We analyze the geometrical vicinity encompassed by each Bresenham's circle $B_r(\mathbf{c})$ in the $3D$ scene by computing the minimum Euclidean distance:

$$\mathbf{d_r} = \min_{p_i} |P(\mathbf{c}) - P(\mathbf{p_i})|, \forall \mathbf{p_i} \in B_r(\mathbf{c}), \tag{6}$$

where $P(\mathbf{c})$ and $P(\mathbf{p_i})$ are the $3D$ points corresponding to the central pixel $\mathbf{c}$ and the pixels composing the Bresenham's circle $\mathbf{p_i} \in B_r(\mathbf{c})$. The distance d_r is weighted by the Gaussian

$$\mathbf{w_r} = \exp\left(-\frac{(\mu - \mathbf{d_r})^2}{\sigma^2} \right) \tag{7}$$

in order to penalize circles which its estimated radii in the 3D scene are distant from $\mu = 0.02$ meters. We then build a feature vector from a Bresenham's circle of radius $\mathbf{r}$ centered at $\mathbf{c}$ as a row vector $\mathbf{v_r} = \left[f_1 \ \ldots \ f_{|B_r(\mathbf{c})|} \right]$ where:

$$f_i(\mathbf{c}, \mathbf{r}) = \mathbf{w_r} * (\tau_v(\mathbf{c}, \mathbf{p_{i,r}}) + \tau_g(\mathbf{c}, \mathbf{p_{i,r}})), \tag{8}$$

where $\mathbf{p_{i,r}}$ is the ith element of the Bresenham's circle $B_r(\mathbf{c})$. The final feature vector $\boldsymbol{F}$ is generated by concatenating all the feature vectors $\mathbf{v_r}$ as, in this work:

$$\boldsymbol{F} = \left[\mathbf{v_3} \ \mathbf{v_5} \ \mathbf{v_7} \ \mathbf{v_9} \right]. \tag{9}$$

Decision Tree Training. In the training step, we create a keypoint model by training a decision tree using the ID3 algorithm [2]. We generated a training set by extracting a total of 160,144 points from the RGB-D Berkeley 3-D Object Dataset (B3DO) [6]. This dataset is composed of a large number of real world scenes with several different objects, geometry and visual data.

We used 66% points to train, and the remaining points (54,449) to test the quality of the final decision tree. Both sets were equally divided into positive and

negative samples. In order to define positive labels for keypoints, we computed the curvature of several, manually selected, keypoints. We have found the value of 0.09 based on the average of these curvatures. Thus, all points with curvature larger than 0.09 were labeled as a positive sample for the keypoint class. To take into account texture features, we added keypoints detected by ORB as positive examples. The classification accuracy obtained in the test was equal to 0.91.

Non-maximal Suppression. In the last step of our methodology we perform a non-maximal suppression. We compute a response value $\mathrm{R_p}(\mathbf{c}, \mathbf{r})$ based on the feature values of each circle. For each radius $\mathbf{r} \in \mathcal{S}$,

$$\mathrm{R_p}(\mathbf{c}, \mathbf{r}) = \max_{\mathrm{X} \in \{\mathrm{X_{rc_1}}, \mathrm{X_{rc_2}}\}} \frac{1}{|\mathrm{X}|} \sum_{x_i \in \mathrm{X}} D_v(\mathbf{c}, \mathbf{x_i}) + \lambda D_g(\mathbf{c}, \mathbf{x_i}), \tag{10}$$

where $D_v(\mathbf{c}, \mathbf{x}) = |I(\mathbf{x}) - I(\mathbf{c})|$ gives the visual response and $D_g(\mathbf{c}, \mathbf{x}) = 1 - \langle N(\mathbf{x}), N(\mathbf{c}) \rangle$ provides the geometrical response. The factor λ is used to define the contribution of the geometrical information in the final response. The set $\mathrm{X_{rc_k}} = \{\mathbf{p_i} : \ \mathbf{p_i} \in B_r(c) \ \wedge \ (\tau_v(\mathbf{c}, \mathbf{p_i}) = k \ \vee \ \tau_g(\mathbf{c}, \mathbf{p_i}) = k)\}$ is composed of all pixels which bin has value k.

We rank the maximal points by using both absolute difference between intensities and normal surface angles for the pixels in the contiguous set of the Bresenham's circle. The final response of each candidate is defined as the maximum response among all radii:

$$\mathrm{R_f}(\mathbf{c}) = \max_r \ \mathrm{R_p}(\mathbf{c}, \mathbf{r}), \forall r \in \{3, 5, 7, 9\}. \tag{11}$$

We divide the image into smaller patches with size $w \times w$ (in this work, $w = 5$) and for each patch we select the pixel with the larger response, Equation 11.

3 Experiments

We compared our approach against standard detectors for two-dimensional images: SIFT [7] and ORB [13], for geometric data HARRIS3D [14] (a 3D version of Harris corner detector), SIFT3D [1], and the HARRIS6D [14]. The HARRIS6D detector, similarly to our methodology, uses both visual and geometrical data to detect keypoints. In our experiments, we used the RGB-D SLAM Dataset [16]. This dataset contains several RGB-D data of real world sequences and for each acquisition it provides the ground truth for the camera pose. For our experiments, we used the sequences containing only translation motions (*freiburg2_xyz*) and rotation of the camera (*freiburg2_rpy*).

To evaluate each detector, we applied the repeatability score, which measures the ability of a detector to find the same set of keypoints on images acquired of a scene from different view points or different conditions. For details the reader is referred to [9]. In our experiments, we used the parameter $\epsilon = 0.6$.

[1] available in the Point Cloud Library: www.pointclouds.org

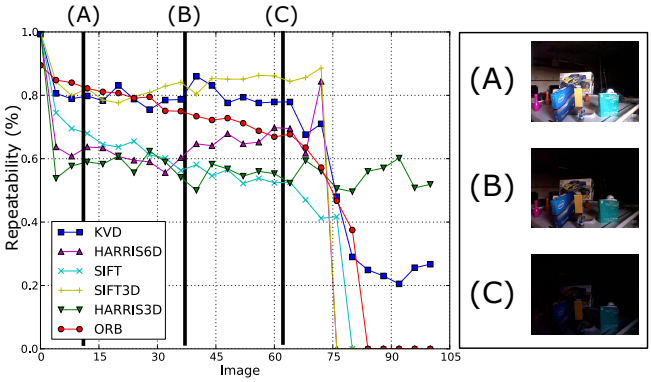


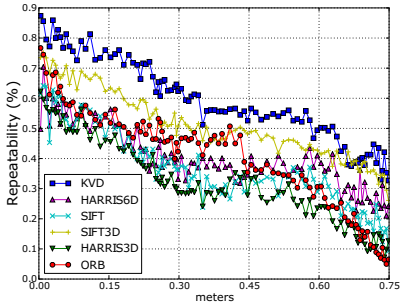
Fig. 2. The poor illumination experiment. We captured a total of 104 images of a cluttered room starting at dusk (on the right three examples of the used images). One can notice that KVD is the only method which uses visual information, that remains providing reliable keypoints once the intensity image is nearly lost (C image).

For the parameter settings, to choose a value for t_g, we ran the learning and testing for 15, 30 and 60 degrees using the RGB-D Berkeley 3-D Object Dataset (B3DO) [6]. We used a fraction of the dataset for validation purposes and the remaining part for training, the large accuracy was returned by using $t_g = 15$.

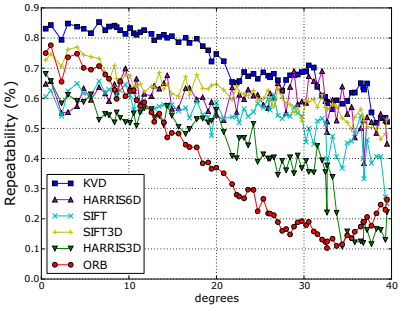
Robustness. We performed experiments to evaluate the repeatability for images acquired with changes in translation, rotation, scale, and illumination. We used offsets ranging from 0.03 meters to 0.75 meters (horizontal direction) for translational tests and for scale tests we select a set of frames with the camera moving away from the scene up to 0.35 meters. In the illumination change experiments we captured a total of 104 images of a cluttered room starting at dusk (partial illumination) at an interval of one minute between acquisitions. Figure 2 shows three frames of this sequence.

Figure 3 shows the results of the repeatability tests. Our detector provides the highest repeatability rate when there are large translational movements (0.8 meters) and large angular rotations (50 degrees). Also in Figure 3 (d),(e) we can see that only KVD, HARRIS6D and HARRIS3D were still capable to provide keypoints from heavily corrupted images under illumination changes and noise, and KVD presented the highest repeatability rate. It is worth noticing that in the illumination change experiment (Figure 2), KVD was the only method which uses visual information that was capable of still provide keypoints after the visual information vanishes (about image 80).

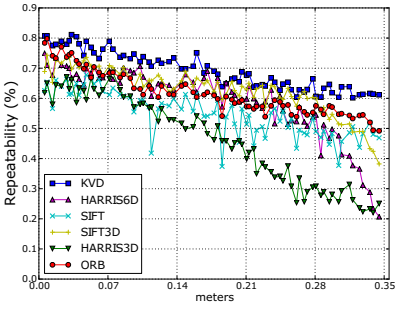
To perform brightness changes, we gradually increased the value of each pixel by adding an increasingly higher constant β using images from *freiburg2_xyz* sequence. To test the robustness to image noise, we used a Gaussian additive noise with zero mean. In Figure 3 one may readily see that our detector presents the largest repeatability rate, thanks to the visual and geometric information fusion of our detection methodology.



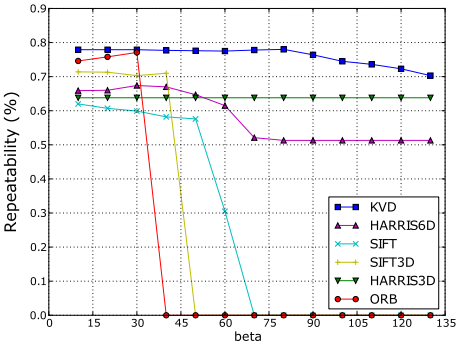
(a) Translational motion



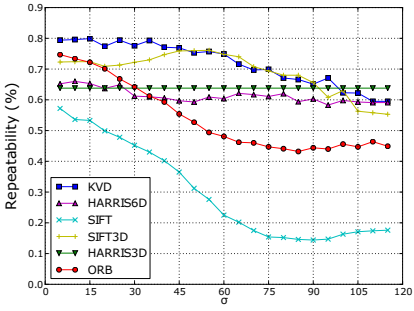
(b) Rotational motion



(c) Scale change



(d) Brightness



(e) Noise

Fig. 3. Results for the repeatability experiment. (a) Horizontal translation camera motion; (b) Rotational movement around the yaw axis; (c) Scale changing; (d) Brightness changing and (e) Gaussian Noise. Our method (KVD) is represented by the blue curve. One can readily see that KVD, among all methods which uses visual information (including HARRIS6D), is the one which continues to identify the most reliable keypoints even when strongly corrupted intensity images are given as input.

Time Performance. The time experiments ran on an Intel Core i7 3.5GHz (using only one core). Time measurements were averaged over 900 runs and over all keypoints. Comparing to other detectors which use geometrical data, KVD was the fastest approach, processing in the order of 10^6 pixels per second, taking 0.06 seconds to process an image of size 640×480 pixels, while its main competitor (HARRIS 6D) takes 0.08 seconds to images of the same size.

4 Conclusion

In this paper we proposed KVD, a novel keypoint detector capable of working with texture and geometrical data. A comparative analysis in terms of robustness to affine transformations was conducted against the standard detectors in the literature for appearance and geometric information and our detector presented the higher repeatability rate.

References

1. Bay, H., Ess, A., Tuytelaars, T., Gool, L.J.V.: Speeded-up robust features (SURF). Computer Vision and Image Understanding **110**(3), 346–359 (2008)
2. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth and Brooks, Monterey (1984)
3. Chen, H., Bhanu, B.: 3D free-form object recognition in range images using local surface patches. Pattern Recognition Letters **28**(10), 1252–1262 (2007)
4. Harris, C., Stephens, M.: A combined corner and edge detector. In: Proceedings of the Alvey Vision Conference, AVC, pp. 1–6 (1988)
5. Holzer, S., Shotton, J., Kohli, P.: Learning to efficiently detect repeatable interest points in depth data. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part I. LNCS, vol. 7572, pp. 200–213. Springer, Heidelberg (2012)
6. Janoch, A., Karayev, S., Jia, Y., Barron, J.T., Fritz, M., Saenko, K., Darrell, T.: A category-level 3-D object dataset: Putting the kinect to work. In: IEEE International Conference on Computer Vision Workshops, ICCV 2011 Workshops, Barcelona, Spain, November 6–13, 2011, pp. 1168–1174 (2011)
7. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision **60**(2), 91–110 (2004)
8. Mikolajczyk, K., Schmid, C.: Scale and affine invariant interest point detectors. International Journal of Computer Vision **60**(1), 63–86 (2004)
9. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Gool, L.J.V.: A comparison of affine region detectors. International Journal of Computer Vision **65**(1–2), 43–72 (2005)
10. Moravec, H.P.: Towards automatic visual obstacle avoidance. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence, p. 584 (1977)
11. do Nascimento, E.R., Oliveira, G.L., Vieira, A.W., Campos, M.F.M.: On the development of a robust, fast and lightweight keypoint descriptor. Neurocomputing **120**, 141–155 (2013)
12. Rosten, E., Porter, R., Drummond, T.: Faster and better: A machine learning approach to corner detection. IEEE Trans. Pattern Anal. Mach. Intell. **32**(1), 105–119 (2010)

13. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.R.: ORB: an efficient alternative to SIFT or SURF. In: IEEE International Conference on Computer Vision, ICCV 2011, Barcelona, Spain, November 6–13, pp. 2564–2571 (2011)
14. Rusu, R.B., Cousins, S.: 3D is here: Point cloud library (PCL). In: IEEE International Conference on Robotics and Automation, ICRA (2011)
15. Steder, B., Rusu, R.B., Konolige, K., Burgard, W.: Point feature extraction on 3D range scans taking into account object boundaries. In: IEEE International Conference on Robotics and Automation, ICRA 2011, Shanghai, China, May 9–13, pp. 2601–2608 (2011)
16. Sturm, J., Magnenat, S., Engelhard, N., Pomerleau, F., Colas, F., Burgard, W., Cremers, D., Siegwart, R.: Towards a benchmark for RGB-D SLAM evaluation. In: Proc. of the RGB-D Workshop on Advanced Reasoning with Depth Cameras at Robotics: Science and Systems Conf (2011)
17. Zaharescu, A., Boyer, E., Varanasi, K., Horaud, R.: Surface feature detection and description with applications to mesh matching. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 373–380 (2009)