Three-dimensional interest point detection and description using Speeded-Up Robust Features and histograms of oriented points

DOI:
10.1109/ICSTCC.2015.7321318

Link to publication record in Manchester Research Explorer

Citation for published version (APA):

Published in:
System Theory, Control and Computing (ICSTCC), 2015 19th International Conference on

Citing this paper
Please note that where the full-text provided on Manchester Research Explorer is the Author Accepted Manuscript or Proof version this may differ from the final Published version. If citing, it is advised that you check and use the publisher's definitive version.

General rights
Copyright and moral rights for the publications made accessible in the Research Explorer are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

Takedown policy
If you believe that this document breaches copyright please refer to the University of Manchester’s Takedown Procedures [http://man.ac.uk/04Y6Bo] or contact uml.scholarlycommunications@manchester.ac.uk providing relevant details, so we can investigate your claim.
Three-Dimensional Interest Point Detection and Description using Speeded-Up Robust Features and Histograms of Oriented Points

Salvador Pacheco-Gutierrez*, Alexandru Stancu
Mohamed Mustafa, Eduard Codres
The University of Manchester
School of Electrical and Electronic Engineering
Sackville Street Building
M13 9PL, Manchester, United Kingdom.
Email: {salvador,pachecogutierrez, alexandru.stancu, mohamed.mustafa, eduard.codres}@manchester.ac.uk

Abstract—This article presents a novel systematic methodology for the detection of interest points in 3D point clouds and its corresponding descriptors by using the information of an RGB camera and a structured-light sensor. This is achieved by fusing Speeded-Up Robust Features (SURF) in the image space, and histograms that statistically represent the relationship of three dimensional geometric data around the interest points. The SURF algorithm is implemented over an image whose pixel coordinates represent values of singular occurrences of the descriptor, thus yielding to particular points on the model. Furthermore, a histogram over the model is calculated and the lower bin values are selected as interest points since they have a direct corresponding 3D point, thus allowing the fusion of both approaches. By combining both methodologies, it is intent to define a set of interest points whose descriptors are able to maintain the intrinsic characteristics of its constituent parts such as repeatability, distinctiveness and robustness while remaining compact and fast to compute. The detected points will be used for both, localization and mapping of mobile robots in partially unknown environments.

Keywords—Computer vision; Robotics; Landmark Detection and Characterization; SLAM.

I. INTRODUCTION

Simultaneous localization and mapping (SLAM) [1], is an approach widely used in mobile robotics to address the problem of mapping an uncharted spatial environment while maintaining the knowledge of the pose of a robot relative to its initial position. Although, the SLAM problem does not possess a single solution, one of the settings for SLAM proposes the assumption that the environment is crowded of landmarks. This landmarks can be defined as well-determined points in the space that can be used as reference for the self-localization of the robot [1]. Furthermore, if the a priori information of the environment is very reduced, the robot will have to deal with the problem of detection and characterization of those landmarks. The present paper is focused on the detection and characterization of such landmarks (that are called interest points) in a 3D space by fusing image and 3D point features.

Due to the interaction with the surroundings, the use of sensors is inherent to the concept of SLAM. In this sense, several types of sensors have been implemented on mobile robots according to their application. Some examples are: sonar [2], laser range finder [3], time of flight cameras [4], and other approaches that combine two or more types of sensors [3], to mention a few. In this work, the sensor used was Kinect from Microsoft®, since the characteristics implemented by its inventors have allowed sensor fusion, this is, image and depth data [5].

A. Related work

In computer vision, more specifically, in image processing, the detection and use of interest points is very popular for many applications such as matching, object and scene recognition and robot navigation, being the latter of special interest in this work. A brief but complete survey of the local feature detector methodologies and implementations can be reviewed in [6]. Similarly, in 3D model processing, another field of computer vision, the interest point detection has been studied lately as well. A brief but complete survey of the local feature detector methodologies and implementations can be reviewed in [6]. Furthermore, a histogram over the model is calculated and the lower bin values are selected as interest points since they represent values of singular occurrences of the descriptor, thus yielding to particular points on the model. Furthermore, although Difference-of-Gaussians (DoG) is deeply studied in image processing [9], this method has been extended to the third dimension. Following this, Lee et al. [12] introduced the concept of mesh saliency by applying a Gaussian-weighted
evaluation of surface curvatures. High values of saliency are detected as interest points. A variation of the above is presented in [13]. Here, the saliency is applied only over the vertex instead of the curvature values, then DoG is calculated over several scales and the interest points are those whose normal displacement with respect to the original position after the Gaussian filter is high enough. Additionally, other 2D methodologies extended for the third dimension are the Harris detector [14], and the Scale Invariant Feature Transform (SIFT) [15]. The methods presented previously do not make use of the sensor fusion capabilities, thus relying only in a certain type of data. This work is based on image detection and characterization of interest points and the association of its geometrical characteristics in the 3D space.

B. Descriptor fusion algorithm

In this paper, we make use of the sensor fusion capabilities of Kinect: the depth and RGB sensors. This is, Kinect allows to find a set of 3D points with coordinates \((x_i, y_i, z_i)\) that have a corresponding \((x, y)\) pixel coordinate in a RGB image, as can be seen in Fig. 1. The relationship between RGB and depth cameras is given by a series of projections and rigid transformations. Thus, is possible to analyze the data from two perspectives: image and point cloud space. First, we focus on the detection and description of interest points over the image space based on the scheme proposed by Bay et al. [7], called Speeded-Up Robust Features (SURF). Once done this, an Euclidean distance similarity criteria is applied to a pair of descriptors corresponding to a scene taken from two different points of view. These are the descriptors corresponding to the interest points \((x, y)_i\) and \((x, y)_j\) in Fig. 1. The resulting interest points are then transferred to their corresponding 3D point cloud positions \((x_i^p, y_i^p, z_i^p)\) and \((x_j^p, y_j^p, z_j^p)\), in order to find a second descriptor based on the methodology proposed in [16]. As a result, a set of interest points that correspond to both scenes is retrieved. In this manner, these points can be used as landmarks for SLAM and at the same time, as anchoring points for point cloud registration.

The paper is organized as follows. In section two, the interest point detection and description methodology for the 2D image is presented. Section three depicts the procedure followed to create the 3D descriptor. In section four, the results of the experiments performed are presented. The paper finishes with a conclusion and outlook of the proposed work.

II. IMAGE ANALYSIS

The first step of the proposed algorithm for the interest point detection consist in analyze the image acquired from the RGB camera. This is a 640 × 480 pixels RGB image. The detector is required to be highly repetitive, in other words, capable to detect the same substantial interest points of a scene regardless changes in the viewing conditions. By analyzing the reliability of several detectors [6], [7], is possible to reach the conclusion that the Hessian-based detectors are more accurate, robust and efficient than their counterparts. Bearing this in mind, the SURF technique proposed by Bay et al. [7], was selected due to the fact that its implementation relies on a Hessian matrix approximation.

A. Interest Points Detection

The detection of the interest point relies on finding \(Q \in \mathbb{R}^2\), such that:

\[
Q = \{ \det(H(p_i, \sigma)) | \forall i \in \{1, \ldots, m\}, p_i \in \mathbb{R}^2, p_i \in I \}
\]

being \(p_i\) the \(i\)th pixel coordinate \((x, y)_i\) over the gray scale image \(I\). The Hessian matrix \(H\) is represented as follows

\[
H(p_i, \sigma) = \begin{bmatrix}
L_{xx}(p_i, \sigma) & L_{xy}(p_i, \sigma) \\
L_{xy}(p_i, \sigma) & L_{yy}(p_i, \sigma)
\end{bmatrix}
\]

being \(\sigma\) the standard deviation that represents the size of the Gaussian distribution. The elements of the Hessian \(H\) can be defined as follows:

\[
L_{xx}(p_i, \sigma) = \frac{\partial^2}{\partial x^2} g(p_i) * I(p_i)
\]

for \(L_{xx}\), and in a similar way for \(L_{yy}\) and \(L_{xy}\). Equation 3 represents the convolution of the second order Gaussian derivative kernel \(g\) with the the image \(I\) at pixel \(p_i\).

Although Gaussians are widely used in scale-space [17], they need to be discretized in order to have a real application. The discretized kernels are called \(D_{xx}, D_{yy}\) and \(D_{xy}\). Due to the square shape of the Gaussian kernel, the rotational invariance property is affected. However, this kernel performs very well in a computational-wise cost-benefit balance [7].

Once lodged the above, the determinant of the approximated Hessian matrix is calculated using the following weighted formula:

\[
\det(H) = D_{xx}D_{yy} - (wD_{xy})^2
\]

In [7], Bay proposes a weighting value \(w = 0.9\) in order to balance the equivalence between the Gaussian kernels and their discretized versions. The resultant set \(Q\), consequence of

Fig. 1. Graphical interpretation of the methodology. Information is gathered from two different positions at time \(t_1\) and \(t_2\) using Kinect. The interest points are represented by black dots and denoted by \((x, y)_i\) and \((x, y)_j\) in the first and second frame respectively. Their corresponding 3D points in the cloud are denoted by \((x_i^p, y_i^p, z_i^p)\) and \((x_j^p, y_j^p, z_j^p)\) respectively.
applying (4) to all points $p_i$, results in the blob response map for the scale $\sigma$. By changing the scale value of the kernel, it is possible to build a stack of blob response maps for different scales.

The scale space is built by increasing the size of the discretized Gaussian kernels in a proportional manner. This is carried out contrary to the traditional method used by Lowe [9] that consists in reduce the size of the image in a pyramidal fashion.

Once created the stack of the blob response maps for several scales ($s$), a thresholding procedure on every scale is yielded in order to remove points under a determined value. Subsequently, each point undergoes a non-maximum suppression procedure in the surrounding neighborhood. This is, the point under analysis is compared to its immediate neighbors located on the lower, upper and actual scale, resulting in a 27-element neighborhood including the point under analysis.

The above lead us to obtain a set of interest points $P_l$ with pixel coordinates $(x_i, y_i)$ for $i = 1, \ldots, l$.

**B. Interest Point Description**

The SURF descriptor is based on the analysis of the intensity values spread around the interest point. This information is analyzed and poured into a feature vector for comparison tasks. Such analysis consists of two main stages: a) determination of the orientation based on the intensity values in a circular neighborhood around the interest point, and b) the analysis of a squared neighborhood aligned with the detected orientation followed by the computation of a vector describing the intensity distribution.

In this work, we make use of the upright version of SURF (coined U-SURF) for the orientation assignment. In other words, we dismiss the computation of the interest points directions. This is due to the fact that in our application, the robots carrying the cameras will be moving in a horizontal plane, thus, the rotation of the camera will occur only around an axis perpendicular to the robot’s moving plane. What is more, even neglecting the orientation assignment, a tolerance of $\pm 15$ degrees is maintained, [7], [8].

As stated previously, the second stage of the description process consists in analyze a squared region centered at the interest point whose size is proportional to the scale ($s$) at which the interest point was detected. Such proportion corresponds to a side length of $20s$. The area is divided into 16 sub-regions with equally spaced sample pixels that are analyzed by means of first order Haar wavelet responses in $x$ and $y$ direction of size $2s$. The response to the Haar wavelets is calculated over all the subregions and a weighting procedure is carried out using a Gaussian filter. In order to build the descriptor, the summed responses of each subregion in the horizontal and vertical direction, as well as its absolute values are poured into an array using the following configuration: $v = [\Sigma d_x, \Sigma d_y, \Sigma |d_x|, \Sigma |d_y|]$. Since we determine 4 values for each subregion, a 64-element vector can be determined. Furthermore, the descriptor is normalized to a unit vector in order to achieve contrast invariance [7].

**C. First Matching Process**

Following the methodology presented in Fig. 1, a first matching process between two sets of interest points and their descriptors is carried out. Those sets represent the detected interest points on a reference and a target image.

The first step to localize the pairs of points that correspond to the same physical location in both scenes is to compare the sign of the trace of the Hessian matrix calculated in (2). The sign differentiates a bright blob from a dark one, allowing a fast first filtering factor. Secondly, we made use of the squared Euclidean distance criteria ($d_E$) for every pair of feature vectors in the reference ($v_r$) and target ($v_t$) sets:

$$d_E(v_r, v_t) = \sqrt{\sum_{j=1}^{n} (v_r(j) - v_t(j))^2}$$

being $n$ the size of the descriptor, in this case 64. Thereafter, the calculated distances undergo a threshold procedure in order to identify the most similar candidate pairs. A set of identified features can be seen in Fig. 4.

**III. POINT CLOUD ANALYSIS**

After the description and thresholding of the interest points sets, some false detected pairs may remain. Thus, taking advantage of the depth data provided by the Kinect sensor, a second descriptor is built over the three-dimensional space.

As stated before, every interest point detected using SURF, posses a corresponding 3D point on the cloud. Hence, the geometrical relations of this 3D interest point can be exploited to gather extra information thus increasing the robustness of the detected point. This methodology consists of the following stages. First, the 3D neighborhood surrounding the interest point is located. Secondly, a statistical representation of the relations between the points in the neighborhood is determined using the methodology proposed by Wahl et al. [16]. Finally, a second matching stage over the already selected interest point pairs is performed to dismiss false positives. As a result, we obtain a set of 3D points in the reference point cloud that have a correspondent point in the target point cloud.

**A. 3D Neighborhood Determination**

The 3D neighborhood determination for every interest point consists of two steps. First a circular neighborhood around the interest point over the 2D image is determined. The above can be expressed as follows:

$$P_{k2} = \{p_m \mid \sqrt{(x_m - x_i)^2 + (y_m - y_i)^2} \leq r \} \quad \forall p_m \in I$$

being $P_{k2}$ the two-dimensional pixel neighborhood, $x_i$ and $y_i$ are the pixel coordinates of the $i$-th interest point under analysis and $r$ the radius of the two-dimensional circular neighborhood. In our tests, a value of $r = 6$ pixels was conveniently selected. Posteriorly, for each pixel in $P_{k2}$ its corresponding 3D point is located. However, some of the points may not represent the actual spatial neighborhood. To counteract this, the points that are farther away than a certain threshold distance are rejected (red points in Fig. 2). Thus, the remaining points are the 3D neighborhood $P_{k3}$ (yellow points in Fig. 2).

Using this simple, but accurate methodology, we avoid the implementation of computational expensive nearest-neighborhood techniques such as Ball Tree, K-d Tree, Principal Axis Tree (PAT), etc.
B. Three-Dimensional Interest Point Description

In order to describe a 3D point is necessary to analyze the geometrical relationships between the points lying in its neighborhood. For this matter, Wahl et al. [16] proposed a compact expression for the relationship between the normals\(^2\) of points in a 3D space. For the normal estimation problem, there are many methodologies and comparisons in the literature, for instance [20]. However, the normal estimation algorithm is not within the scope of this paper.

The graphical description of the parameters analyzed in every pair of oriented points\(^3\) can be observed in Fig. 3.

\[
\begin{align*}
\alpha &= \frac{u \cdot w}{n_1} \\
\beta &= \frac{v \cdot n_2}{n_2} \\
\gamma &= \frac{u \cdot (p_2 - p_1)}{n_2} \\
\delta &= \frac{|(p_2 - p_1)|}{n_2}
\end{align*}
\]

which are the values that define the so called 3D feature vector \(F = (\alpha, \beta, \gamma, \delta)\).

The relationships expressed in (10)–(13) are applied to all the pairs of oriented points within the \(K_3\) neighborhood. This corresponds to a number of \(k(k - 1)/2\) features per interest point. These features are then binned into a four dimensional histogram with 5 bins per attribute and reshaped in a linear vector of size \(n^a\) being \(n\) the number of bins per attribute and \(a\) the number of attributes on each feature, in this case 5 and 4, respectively.

In order to determine the range over which the bins will be split, is necessary to analyze the nature of the attributes expressed in (10)—(13). First, is easy to note that \(\alpha\) is calculated using the atan2 function, thus its values will fall in the range \([-\pi, \pi]\]. The values for \(\beta\) and \(\gamma\) are estimated from the projection of unit vectors, thus their values fall between \([-1, 1]\]. Finally, the parameter \(\delta\) can be bounded to the diameter of the sphere that determines \(K_3\), this is \([0, d]\). Once the histogram is completed, a vector of size 625 is determined for each interest point.

C. Second Matching Process

A second thresholding stage is implemented to dismiss the false positives detected by the SURF algorithm. To do so, an Euclidean distance similarity criteria is applied to the pair of 3D feature vectors that correspond to all the pairs of interest points detected in Section II-C.

IV. RESULTS

In this section we present the results of the proposed methodology. All the algorithms discussed in this work were implemented in Matlab\textsuperscript{®} R2013a. The data analyzed was taken from a simple office scenario from two different perspectives, including rotation and translation. Rotation is held according to the requirements of U-SURF (see section II-B). The two images were analyzed and the interest points detected can be seen in Fig 4.

\[\text{(a)} \quad \text{(b)}\]

Fig. 4. Interest points detected by the SURF algorithm. The size and color of the circles surrounding the interest points are selected according to the scale they were detected.

Following the procedure, a first matching process between

---

\(^2\)The normal of a point is the vector perpendicular to the surface where the point lies.

\(^3\)An oriented point can be defined as a point with a certain orientation determined by its normal.
the two sets is applied with different threshold values for the first rejection stage, \( t_s \). In Fig. 5, a plot of detected points versus the percentage of false positive detections is presented. As can be seen, decreasing \( t_s \) represents a reduction of the false positives but also a reduction in the detected points.

![Fig. 5. False positive rates against point detection. The numbers next to each marker represent the threshold \( t_s \) used for the first Euclidean distance rejection.](image)

Once the first sets of interest points are detected, the point cloud analysis presented in section III is carried out. Said procedure is performed for different sets of interest points detected with different \( t_s \) values, such values can be observed next to the markers in Fig. 5. Thus, for each pair of matched interest points in the image space with a threshold value \( t_s \), a second rejection procedure is executed, but this time we make use of the descriptors retrieved in section III-B and with a set of thresholds \( t_H \). The results can be seen in Fig. 6.

![Fig. 6. False positive rates against point detection for the second rejection stage. Each line represents a different \( t_s \) used in the first matching stage. The markers on every line represent a Euclidean threshold value \( t_H \) of 0.5, 0.45, 0.40, 0.35, 0.30, 0.25 from left to right.](image)

As can be seen in Fig. 5—6, the false positive detected rates decrease after the second rejection without losing many authentic positives. We can consider the second descriptor as a fine adjustment of the SURF detector. What is more, in Fig. 6, it is possible to observe that values for \( t_H \) less than 0.3, rather than decrease false positives, eliminate authentic positives. So we could set \( t_H = 0.3 \) as the lowest threshold value.

As an example, in Fig. 7, a first set of interest points detected using SURF is presented. Then, in Fig. 8, the resulting set of interest points after feature histogram rejection can be observed. To conclude, this final set can be brought to the three-dimensional plane as observed in Fig. 9.

![Fig. 7. Interest points detected in both scenes after first matching stage using the SURF descriptors. The threshold value for the Euclidean distance rejection procedure was \( t_s = 0.45 \).](image)

![Fig. 8. Interest points detected in both scenes after second matching stage using the feature histogram descriptors. The threshold value for the Euclidean distance rejection procedure was \( t_H = 0.30 \).](image)

![Fig. 9. Resulting three-dimensional interest points. The points in white represent the corresponding 3D points in the cloud for each set.](image)

A. Application to the estimation of robot position by means of rigid transformation

The presented algorithm was implemented in [21]. In this work, the rigid transformation between different locations (see Fig. 1) of a Pioneer P3-DX mobile robot is calculated using Interval Analysis (see Fig. 10). The robot is moving around a static scenario while detecting landmarks at different positions. The problem to be solve is the following: given two sets of landmarks taken from two different positions, \( X_1 = \{x_1^1, x_1^2, \ldots, x_1^m\} \) and \( X_2 = \{x_2^1, x_2^2, \ldots, x_2^m\} \), being \( x_i^1, x_i^2 \in \mathbb{R}^{2 \times 1} \). Is needed to estimate the translation \( t \) and rotation \( R \) that satisfy the equation:

\[
x_b^i = RT (x_a^i - t), \quad i = \{1, \ldots, m\}
\]

where \( t \in \mathbb{R}^{2 \times 1} \), \( R \in \mathbb{R}^{2 \times 2} \), and \( \text{Det}(R) = 1 \). The above performs the rigid transformation of the robot motion from
position $i−1$ to $i$. The results can be observed in Fig. 10. Since the robot motion is constrained to be in two dimensions, the Z coordinate of the landmarks is neglected and only the X and Y position is considered.

Fig. 10. Mobile robot moving in a static scenario. The robot actual path is represented with the black line. The estimated position is represented by red boxes. The blue stars are the X and Y coordinates of the detected landmarks.

V. CONCLUSIONS

In this paper, a fusion of two types of descriptors has been presented. This task was achieved using the sensor fusion capabilities of Kinect. First, the image data has been analyzed using Speeded–Up Robust Features for detection and description of interest points. A first matching stage was performed to detect corresponding points from a second image of the same scene but taken from a different perspective. The resulting matched pairs of points have been translated to their 3D counterpart for geometrical analysis. By analyzing the geometrical relationships between the points lying in the 3D neighborhood, it was possible to refine the previously detected interest points. The above has increased the reliability of the interest points by decreasing the false positive detection rate. The results obtained in this paper will be used for two main purposes, the self-localization of a mobile robot in a partially unknown environment and for mapping. For the former, new methodologies focused on interval methods are currently under research in our group. For the latter, point cloud registration and model identification techniques will be implemented [22],[23].

ACKNOWLEDGMENT

The main author would like to thank The National Council for Science and Technology (CONACyT) for funding his PhD studies. The authors would like to acknowledge the support of the funding bodies and industrial partners who have supported the work presented in this paper including the Innovate UK, Sellafield Ltd, the National Nuclear Laboratory (NNL), Dalton Nuclear Institute (DNI) and the UK Nuclear Decommissioning Authority (UKNDA).

REFERENCES


