

ASN: Image Keypoint Detection from Adaptive Shape Neighborhood

Jean-Nicolas Ouellet and Patrick Hébert *

Computer Vision and Systems Laboratory
Laval University
Quebec, QC, Canada, G1V 0A6
{jouellet,hebert}@gel.ulaval.ca

Abstract. We describe an accurate keypoint detector that is stable under viewpoint change. In this paper, keypoints correspond to actual junctions in the image. The principle of ASN differs fundamentally from other keypoint detectors. At each position in the image and before any detection, it systematically estimates the position of a potential junction from the local gradient field. Keypoints then appear where multiple position estimates are attracted. This approach allows the detector to adapt in shape and size to the image content. One further obtains the area where the keypoint has attracted solutions. Comparative results with other detectors show the improved accuracy and stability with viewpoint change.

1 Introduction

Recovering the accurate position of a camera from visual features usually involves placing artificial targets in the scene to ensure the detection and accurate localization of reliable keypoints from multiple viewpoints. To improve flexibility, natural features that are present in the scene should contribute. A critical aspect is the selection and position estimation of such reliable image keypoints especially when their number is limited. Each keypoint is important and no optimization procedure will be adequate unless sets of image keypoints accurately correspond to the same physical points in the scene.

Interest point, or keypoint, detection has been widely studied over the last 30 years and several detection approaches have been proposed. These points are distinctive locations in the image such as corners, other types of junctions and blob regions. A good review is presented in [1] where the authors study the repeatability of an observation under several types of transformations including severe scale and photometric changes.

In the last 10 years, significant progress has been accomplished in coupling multi-scale feature detectors with descriptors. This has been motivated by finding point correspondences between two images or more generally in model-based recognition [2]. From the literature, it appears there is currently no universal detector/descriptor but interestingly, a combination of complementary operators seems to be a reasonable solution. While rich feature point descriptors will help in obtaining good initial correspondences, accurate keypoints that are only recognizable locally based on their seldomness will be used for improving the accuracy. We believe there is no compromise to be made

* The authors are grateful for support from NSERC Canada

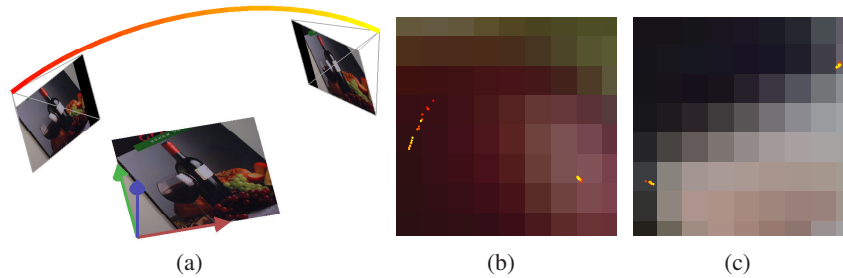


Fig. 1. Tracking the displacement of image features with viewpoint. a) Image features are detected along a trajectory over a rotation of 90 degrees. A dot is marked in the planar scene with its color corresponding to the camera position along the trajectory. b),c) Points backprojected onto the scene for the extracted SIFT and ASN approaches respectively.

between obtaining keypoints associated with discriminant local descriptors and accurate keypoints. When the number of accurate keypoints is not sufficient, one will have to add artificial targets. In this context, we focus on the localization accuracy of keypoints under viewpoint change. Although there may be severe viewpoint changes due to rotation, severe image scale changes are less an issue in this work. Moreover, in order to decouple the effect of occlusion, our analysis is limited to planar scenes.

An illustrative example is presented in Fig.1. While some SIFT features [3] are very stable, as shown on the right of Fig.1(b), others, such as the point on the left, will exhibit a gradual displacement reaching a few pixels as the camera moves along a trajectory. The displacements observed are related to the local image structure configuration which affects the stability under perspective distortion. In this simulation with a real image, the detected points are mapped in the fronto-planar view. The colors of the feature points correspond to the camera position along the trajectory. For applications like object recognition, the presence of such a displacement may not significantly affect the performance or quality of the results. However, this is not the case when the features are used to recover the exact geometry of the scene.

The classic direct approach for identifying image keypoints consists in applying a detector that evaluates a function within the neighborhood of each pixel in an image. Then, local extrema are identified, validated and their position can be refined based on the function value, for subpixel precision. The function is typically based on the local estimation of the second moment matrix of the gradient or the Hessian matrix [4, 5]. The relative accuracy of this type of detectors was studied for an ideal corner model [6, 7]. Nevertheless, in images of real scenes the result will be affected by the presence of several structures within a neighborhood. Moreover, when varying the viewpoint, the projective transformation will affect the distance between local structures in the image and an estimate based on a fixed size and shape neighborhood will be affected. Methods for adapting the neighborhood have been proposed for improving descriptors but this does not improve accuracy [4, 8]. For localization accuracy, the shape of the neighborhood should reflect the image content.

We propose to reconsider the detection scheme. Instead of applying a detector, we systematically estimate the point of convergence of the local gradient field within the neighborhood of each pixel in the image. These potential keypoints can be located several pixels away from the center of their estimation neighborhoods. As the detector processes the image, certain positions will attract more estimates than others, resulting in the formation of poles. The strength of a pole is the size of the area of attraction. The position of strong poles are then extracted as keypoints. Fig.2 illustrates this idea.

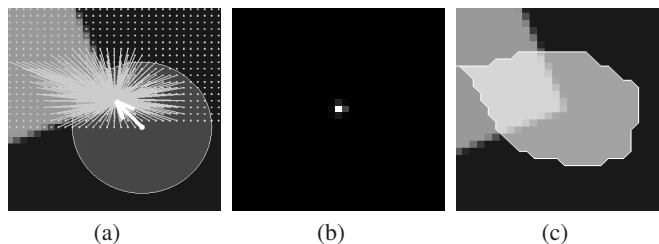


Fig. 2. The ASN detector identifying poles and support map. a) The estimates are obtained at each image position and stored in an accumulator. The final state of the accumulator is depicted in (b) and the corresponding support map is displayed in (c).

This approach is advantageous for three reasons. First, the area for estimating a keypoint adapts in shape based on the image content and needs not to be centered nearby the keypoint. Secondly, although it is possible to further characterize the strength of a pole, the detection is robust to the presence of nearby structures and is fundamentally independent of the image contrast. Finally, the area of attraction may provide further insight on the keypoint interest.

In section 2, definitions are provided and the approach is motivated from the most related works. In section 3, the method and the corresponding algorithm are developed. It is also shown how to apply multiple window scales and how to characterize and validate poles. Experiments follow in section 4 where it is shown that the extracted keypoints are highly stable even with strong viewpoint changes.

2 Keypoint Detection and Localization from the Gradient Field

Currently, most research efforts on keypoint detection focus on the correspondence between images, based on local information. A keypoint is located at the center of a region that is rich for description. It is then important to normalize scale and local deformation due to viewpoint. From the work of Lindeberg [9], who introduced the automatic scale selection to more recent works of Lowe on Scale Invariant Feature Transform (SIFT), Triggs [10] who generalizes the Harris operator to provide repeatable scale and orientation, Mikolajczyk and Schmid [4] who put forward the Harris (Hessian) affine detector, Tuytelaars and Van Gool [8] who proposed edge-based and intensity based detectors, the literature is abundant. Other detectors such as the MSER and salient region detector

[11, 12] are also worth noting. Good reviews on the detectors and descriptors have been published [1, 13] along with other comparisons and evaluations [14, 15].

In this paper we focus on local keypoint detectors. Among these detectors, most of the methods are based on the local measurement of the image Hessian matrix or the second moment matrix of the gradient. Analytical studies for the accuracy of ideal isolated corners can be found in [6, 7]. Among these detectors, the Forstner operator including variants such as the well known Harris detector are based on the second moment matrix [16, 17]. They are particularly interesting since they make it possible to detect various types of keypoints such as L -corners, Y , T , and X junctions. A short analysis of these classic detectors will motivate the new approach proposed in this paper.

Given a point $\mathbf{x} = (x, y)$ in an image I , the gradient $\nabla I(x, \sigma_d)$ at derivation scale σ_d is defined as $\nabla I(x, \sigma_d) = (I_x, I_y)^T$. The components I_x and I_y are obtained by convolving the Gaussian first order derivative $\frac{\partial}{\partial x}g(\sigma_d)$ with the image I at point \mathbf{x} along the x and y directions respectively. The gradient field is the set $\mathcal{G} = \{\nabla I(\mathbf{x}, \sigma_d)\}$ for all \mathbf{x} in the image. The second moment matrix $\Gamma(\mathbf{x}, \sigma_d)$ in \mathbf{x} at derivation scale σ_d is defined as follows

$$\Gamma(\mathbf{x}, \sigma_d) = \begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix}. \quad (1)$$

In order to obtain a local estimate $\bar{\Gamma}$ of Γ , the rotationally symmetric Gaussian function with scale σ_I is typically used for integrating the local gradient field components at point $\mathbf{x}_0 = (x_0, y_0)$

$$\bar{\Gamma}(\mathbf{x}_0, \sigma_d, \sigma_I) = \int \Gamma(\mathbf{x}, \sigma_d) g(\mathbf{x} - \mathbf{x}_0, \sigma_I) d\mathbf{x}. \quad (2)$$

In practice, the integral can be computed over a limited neighborhood window $N(\mathbf{x}_0, s)$ with center \mathbf{x}_0 and size s . From the 2×2 symmetric matrix $\bar{\Gamma}(\mathbf{x}_0, \sigma_d, \sigma_I)$, the eigenvalues can be analyzed or more efficiently, a combination of the trace and determinant of this matrix makes it possible to detect a keypoint by simply applying a threshold on the obtained scalar value. This operation is used for validating the presence of a keypoint in several detectors. For those detectors based on the Hessian matrix \mathcal{H} , the principle is the same except that \mathcal{H} is computed instead of Γ . The maximum values can be further interpolated for subpixel precision of the position.

In [16], Forstner proposes a different way for refining the position estimate \mathbf{x}_0 . To do so, it is proposed to minimize the weighted sum of the squared distances $d(\mathbf{x}_0, \mathbf{x})$ of the reference points \mathbf{x}_0 to the line passing through \mathbf{x} in the direction orthogonal to $\nabla I(x, \sigma_d)$, where the weights are the squared gradient magnitude

$$\hat{\mathbf{x}}_0 = \operatorname{argmin}_{\mathbf{x}_0} \int d^2(\mathbf{x}_0, \mathbf{x}) \|\nabla(\mathbf{x}, \sigma_d)\|^2 g_{\sigma_I}(\mathbf{x} - \mathbf{x}_0) d\mathbf{x}. \quad (3)$$

One can easily assess the benefit of this estimation step when observing an L -corner from different viewpoints. Actually, the opening angle will vary in the image. This behavior was studied by Rohr [6] for the raw detectors without refinement. Fig.3(a) shows the improvement after refinement. This refinement is particularly interesting since the

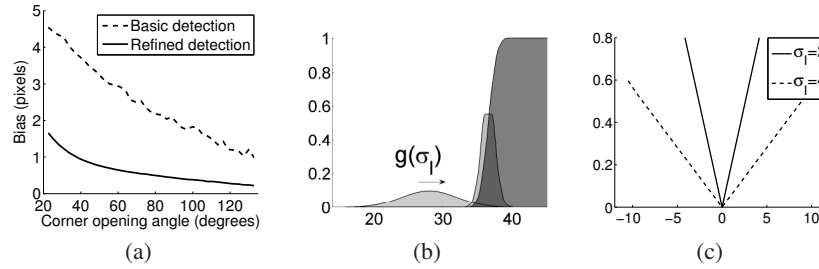


Fig. 3. a) Distance between the detected point and the actual junction with respect to the corner's opening angle before and after refinement of the detection using the Forstner operator. b) Location bias introduced by the Gaussian weight. When not exactly centered on the edge, the Gaussian imposes asymmetric weights on each side of the edge. c) The resulting bias in pixel with respect to the distance between the Gaussian center and the solution.

keypoint location needs not to be at the center of the estimation window N . In practice, we have observed displacements higher than a pixel.

Due to band-limited images and noise, the gradient field is disrupted nearby a junction. Fig.4 illustrates the tangent field for both X and L types of junctions. Here, the tangent field is simply the set of vectors that are orthogonal to the gradient. The tangent field is parallel to edge directions. In Fig.4 the color encodes the gradient magnitude. When the tangent field is symmetric around a junction, the error of the position estimate will be significantly reduced especially when N is centered at the junction. This characteristic is exploited in photogrammetric targets. However, in real scenes where it is more likely to observe L-corner types of junctions, the error will depend more on the integration scale. The larger the scale s , the closer the estimate will be toward the junction. It is then advantageous to select a larger scale. Nevertheless, the more likely N will overlap other structures.

Before completing this section, we consider the effect of the Gaussian weight g_{σ_I} for the integration window at the refinement step. For L -corners it will bias the position toward the interior of the corner. Actually, for this type of corner the maximum response of all operators lies inside the corner [6]. It is easy to understand this effect in one dimension. For instance in Fig.3(b) a step edge is depicted along with its derivative response superimposed and centered on the edge. Then suppose the position of the

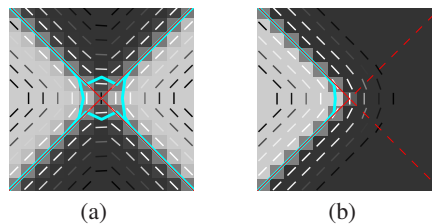


Fig. 4. Measured tangent field for a) an 'X' and b) an 'L' type of junction.

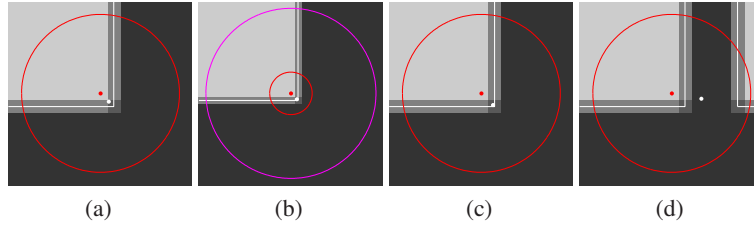


Fig. 5. Corner localization using a) the Forstner operator with $\sigma_d = 2, \sigma_I = 2$. In b) a larger integration scale is used $\sigma_d = 2, \sigma_I = 4$. c) The same window sizes as in (a) are used but without Gaussian weight for the integration window. d) Influence of a nearby structure.

edge is estimated based on the center of gravity of the derivative response but using an integration window with Gaussian weight g_{σ_I} that is offset with respect to the derivative response. In the depicted case, the estimate will obviously be biased toward the left. As shown in Fig.3(c), the bias varies with g_{σ_I} and the offset.

This effect is also observed in images. In Fig.5, one can observe the detection corresponding to the maximum response of the Forstner operator (red point) and the refined position (white point). Increasing the integration scale will reduce the bias (see Fig.5(b)). As expected, if the Gaussian weight is applied for the detection but not for the position refinement, the final result is unbiased (see Fig.5(c)). Nevertheless, that creates another problem depicted in Fig.5(d). The presence of nearby disrupting structures introduces a bias in the estimate. The Gaussian weight used in most keypoint operators greatly contributes to minimize this effect, especially for square windows. It will, however, introduce the bias especially for the common L-corners.

From these observations and analysis, we will retain three conclusions:

1. From the gradient (tangent) field, it is possible to estimate a local optimal position of any type of junctions such as X , T , Y and L -corners.
2. When the estimation is performed away from the keypoint position, a uniform weight in the integration window will prevent bias.
3. A larger integration scale leads to better estimates but is more prone to disrupting nearby structures.

3 Keypoint Detection Based on Pole Identification

Based on these considerations, we propose the following approach to detect and estimate the positions of the keypoints. In order to detect reliable keypoints, we search for poles in the image. There is a pole at a given position \mathbf{x}_0 in the image when multiple estimates from different positions of N are attracted at this position. The number of estimates at a pole is a local peak. The strength of a pole located at \mathbf{x}_0 is the number of window neighborhoods, $N(\mathbf{x}, s)$, that led to \mathbf{x}_0 . It is therefore proposed to systematically apply the least-squares estimate of equation 3 at all positions \mathbf{x} in the image and identify poles without any pre-detection. Significant poles are then validated and

identified as significant keypoints. This approach is remarkably simple in its principle. However, it fundamentally differs from the process of searching for the position where a function of the second moment matrix or Hessian matrix leads to a maximum. Actually, a pole arises from the estimation of neighborhood windows at multiple positions. That is why it can adapt in shape to the presence of nearby structures.

In order to formalize this idea, we revisit the definition of an estimate from eq. 3 with

$$\hat{\mathbf{p}} = F_N(\mathbf{x}, s) = \operatorname{argmin}_{\mathbf{p}} \sum_{y \in N(\mathbf{x}, s)} d^2(\mathbf{p}, \mathbf{y}) \|\nabla(\mathbf{y}, \sigma_d)\|^2. \quad (4)$$

In this equation, we have replaced the integral by a discrete sum over the neighborhood $N(\mathbf{x}, s)$ and we have removed the Gaussian weight in eq. 3. This will eliminate the bias source described in section 2.

Let us also define the support map for a given position as

$$S(\mathbf{y}) = \{\mathbf{x}\} \mid \|F_N(\mathbf{x}, s) - \mathbf{y}\| < \epsilon. \quad (5)$$

$S(\mathbf{y})$ is the set of positions \mathbf{x} whose estimates $\hat{\mathbf{p}}$ fall in the vicinity of \mathbf{y} . There is a pole in \mathbf{y} if the following condition is met

$$\#S(\mathbf{y}) > \#S(\mathbf{y}'), \forall \mathbf{y}' \mid \|\mathbf{y} - \mathbf{y}'\| < \epsilon, \quad (6)$$

where $\#$ stands for the cardinality. The exact subpixel position of the pole is estimated by averaging all estimates $\hat{\mathbf{p}}$ that have been attracted by the pole. In the same way it is accomplished for validating keypoints based on the second moment matrix, it is possible to characterize the second moment matrix of the augmented support map S^+ of a pole. S^+ is defined as the union of all neighboring windows whose estimates have been attracted by the pole

$$S^+(\mathbf{y}) = \bigcup N(\mathbf{x}, s) \mid \mathbf{x} \in S(\mathbf{y}). \quad (7)$$

Fig.2 illustrates the progression of the estimation process along with the attracted estimates and the final support map of a detected pole. It is worth noting that the identification of a pole is not directly related to the junction contrast but to the area of its support map. Moreover, this process will improve robustness similar to Hough-based approaches where a consensus is sought locally [18]. However, we avoid the detection of numerous false points that must be discarded a posteriori [18], by searching for a concentration of independent estimates.

Despite its capability of adapting to the local structure, it is possible that a large integration scale makes it impossible to identify a pole since N always overlaps multiple structures in the image. It is thus mandatory to reduce the integration scale. In order to cope with this situation, the integration scale is reduced progressively. Poles are detected and validated at the largest scale for higher immunity to noise. Then, the scale is recursively decreased and valid poles are sought for at each scale. A small image area around a pole is deactivated such that no new pole can be accepted within the area at smaller scale. The whole procedure is simple and its detailed implementation is described in section 3.2.

3.1 Pole Validation

The number of estimates corresponds to the surface (cardinality) of a pole's support map, S , whose size and shape will vary with the type of junction and the presence of structures in the vicinity of the keypoint. Moreover, the probability that a junction be within the neighboring window $N(\mathbf{x}, s)$ is proportional to the area of the window. Thus a reasonable threshold for validating the strength of a pole is $\tau = 0.2 \text{Area}(N(\mathbf{x}, s))$. This efficiently identifies significant poles while eliminating poles resulting from noise.

For stability, it is not sufficient to reject poles below τ . Unstable poles may appear along low curvature edges. The second moment matrix is computed over the extended support map S^+ and the ratio of its eigenvalues $\lambda_{max}/\lambda_{min}$ is tested. In all cases, the pole is rejected when the ratio of the eigenvalues is higher than 10. Such test is common in the literature; it eliminates an operator response over curved edges [3, 17].

From the second moment matrix, one can further assess the uncertainty of the pole's position. For that purpose, let us reformulate the least-squares solution of eq. 4 into matrix form. The estimates result from the least-squares solution of $\mathbf{K}\mathbf{p} = b$

$$\mathbf{K} \equiv \begin{bmatrix} \frac{\partial I}{\partial x_1} & \frac{\partial I}{\partial y_1} \\ \vdots & \vdots \\ \frac{\partial I}{\partial x_i} & \frac{\partial I}{\partial y_i} \\ \vdots & \vdots \\ \frac{\partial I}{\partial x_n} & \frac{\partial I}{\partial y_n} \end{bmatrix}; b \equiv \begin{bmatrix} x_1 \frac{\partial I}{\partial x_1} + y_1 \frac{\partial I}{\partial y_1} \\ \vdots \\ x_i \frac{\partial I}{\partial x_i} + y_i \frac{\partial I}{\partial y_i} \\ \vdots \\ x_n \frac{\partial I}{\partial x_n} + y_n \frac{\partial I}{\partial y_n} \end{bmatrix}, \quad (8)$$

where n is the number of pixels in $N(\mathbf{x}, s)$. The solution of this system is given by

$$\hat{\mathbf{p}} = F_N(\mathbf{x}, s) = \bar{T}^{-1} \mathbf{K}^T b, \quad (9)$$

with the second moment matrix $\bar{T} = \mathbf{K}^T \mathbf{K}$. A solution exists if \bar{T} is invertible. The uncertainty of the solution can be obtained from the covariance matrix of the point estimate as the product of the error variance with \bar{T}^{-1} , $\mathbf{C}_p = \sigma_{err}^2 \bar{T}^{-1}$. The error variance is given by $\sigma_{err}^2 = \frac{\|\mathbf{K}\mathbf{p} - b\|^2}{n-2}$ [16]. From this uncertainty expression, one can further validate the precision of the poles with the variance of the estimation error, σ_{err}^2 . The variance depends on the noise level as well as on the discrepancy of the local tangent field. As the support map overlaps other contrasted structures nearby, the variance increases whereas the uncertainty will remain low. Poles arising from the interaction with such structures that do not intersect at a single point, are referred to as virtual junctions. Virtual junctions are not located on actual junctions of the image and exist only for specific integration scale. To improve the quality of the points, we decouple σ_{err}^2 from \mathbf{C}_p and eliminate poles for which σ_{err} is higher than a tolerance T_{err} .

3.2 Implementation

In order to obtain the tangent field values at each pixel, the image gradient is first evaluated by convolving the image with a Gaussian derivative filter with $\sigma_d = 1$. Next, using Eq. 9, the least-squares estimate is computed within the integration window $N(\mathbf{x}, s)$ for

all positions in the image. If present, color channels are directly handled by the operator by combining the gradient vectors from the three channels. In our implementation, the image is processed three times with a circular integration window of decreasing radii $s = \{9, 6, 3\}$.

As the operator is applied at each image position, the estimates are stored in the cells of an accumulator. The accumulator’s discretization is the same as the image; this provides a direct correspondence between cells and pixels. Since an estimate will generally not map to the center of a cell, we increment the four closest cells by distributing the coefficients of a bilinear interpolation. This amounts to adding the value of a 2D normalized Gaussian function of $\sigma_A = 0.5$ centered on the estimate. The position \mathbf{x} of N leading to a given estimate is also stored to allow fast extraction of the support map for a given pole. More precisely, we store the list of positions \mathbf{x} at the rounded coordinates of their estimates $\hat{\mathbf{p}}$ in a first table as $\mathbf{H}(\lfloor \hat{\mathbf{p}} \rfloor) \leftarrow \mathbf{x}$ and, in a second table \mathbf{E} , we store each estimate $\hat{\mathbf{p}}$ at the position at which it was estimated, $\mathbf{E}(\mathbf{x}) \leftarrow \hat{\mathbf{p}}$.

After processing the image at a given integration scale s , the local maxima of the accumulator are identified. Positions whose support map contains a sufficient number of estimates ($> \tau$) are identified as potential poles. To extract the support map, one must identify the estimates mapping within $\epsilon = 2 * \sigma_A$ pixels of the local maxima.

The poles are finally re-estimated as the weighted mean of the estimates associated with its support map. A normalized Gaussian weighting function with ($\sigma = \sigma_A$) centered on the local maximum is used for this purpose. The poles are then validated according to the ratio $\lambda_{max}/\lambda_{min} < 10$ and $\sigma_{err} < 0.25$ described in the previous section. As the operator integration scales are explored, poles detected at larger integration scales are prioritized; we reject any local maximum obtained at smaller scales within a radius of $\rho = 2.5$ pixels of a valid pole. The complete process is summarized in algorithm 1.

4 Results

First, we present results with both synthetic and real images to illustrate how the operator can adapt in scale and shape to image content. Then, following the demonstration in Fig.1, we present comparative experiments on the displacement and repeatability of keypoints using virtual cameras to simulate different controlled viewpoints from real images. We compare three detectors: ASN, Forstner with refinement, and SIFT. The images are generated using homographies so the ground truth allows any bias to be identified while preventing matching errors.

4.1 Adaptation to Image Content

In order to illustrate the behavior of the detector, an image was synthesized with squares of different sizes and gaps in between. The corners of the squares, visible at the intersections of the white lines in Fig.6(a), are only detectable at specific scales. The support maps S detected by the ASN detector are superimposed on the image in Fig.6(a) along with the poles identified by white dots. The color of the regions encodes the integration scale at which a pole was identified (blue $s = 9$, green $s = 6$, red $s = 3$ pixels).

```

Input: Image gradient
Output: Poles and Support Maps
foreach Scale  $s$  in  $\{9, 6, 3\}$  do
  Initialize accumulator  $\mathbf{A} \leftarrow \mathbf{0}$ 
  // Estimation
  foreach image position  $\mathbf{x}$  do
    Estimate  $\hat{\mathbf{p}} \leftarrow f_N(\mathbf{x}, s)$ 
    if  $\hat{\mathbf{p}}$  exists then
      Increment the accumulator  $\mathbf{A}(\hat{\mathbf{p}})$ 
      Store  $\mathbf{H}(\lfloor \hat{\mathbf{p}} \rfloor) \leftarrow \mathbf{x}$  and  $\mathbf{E}(\mathbf{x}) \leftarrow \hat{\mathbf{p}}$ 
    end
  end
   $\Omega \leftarrow$  positions of local maxima in the accumulator  $\mathbf{A}$ 
  // Validation
  Remove positions from  $\Omega$  that are closer than  $\rho$  pixels from higher scale poles
  foreach position  $\mathbf{x}$  in  $\Omega$  do
    Extract the support map  $S(\mathbf{x})$  using  $\mathbf{H}$  and  $\mathbf{E}$ 
    if  $\#S(\mathbf{x}) > \tau$  then
      Re-estimate the pole position  $\hat{\mathbf{p}}$  from the estimates in  $\mathbf{E}$ , indicated by  $S(\mathbf{x})$ 
      Compute  $\bar{T}$  over  $S^+(\mathbf{x})$ 
      if  $(\lambda_{max}/\lambda_{min} < 10)$  and  $(\sigma_{err} < T_{err})$  then
        |  $Poles \leftarrow \{\hat{\mathbf{p}}, S(\mathbf{x})\}$ 
      end
    end
  end
end

```

Algorithm 1: The ASN detector: poles and support maps extraction.

The red dots mark the poles (virtual junctions) that were rejected due to high σ_{err} ($\sigma_{err} > 0, 25$). The cases presented in Fig. 6(b) to 6(g) originate from real images. For these latter cases, S^+ is superimposed using color transparency.

A pole will be close to the center of S when there is no other structure in the vicinity. It is also worth noting that a pole is not confined to its support map (see Fig.6(a) and Fig.6(c)). This justifies the importance of estimating a position that may not correspond to the center of the integration window. However, if the image local structure interferes with the operator for all positions near a junction, the estimates might form a strong pole where no real junction exists. Such virtual poles can be observed as red dots in the synthetic example of Fig.6(a), and also for a real image in Fig.6(e). These poles are sensitive to the scale of the operator. They are identified by the analysis of the RMS error obtained after the estimation. At smaller scales the real junction will eventually be identified. In the synthetic example of Fig.6(a), this resulted in the green and red regions corresponding to regions extracted at smaller scales. For real images, the virtual pole in Fig.6(e) was rejected and the real junctions in Fig.6(f) and Fig.6(g) were identified afterwards. Poles can also appear on low curvature edges as displayed in Fig.6(d). These less stable points do not pass the verification on the eigenvalue ratio.

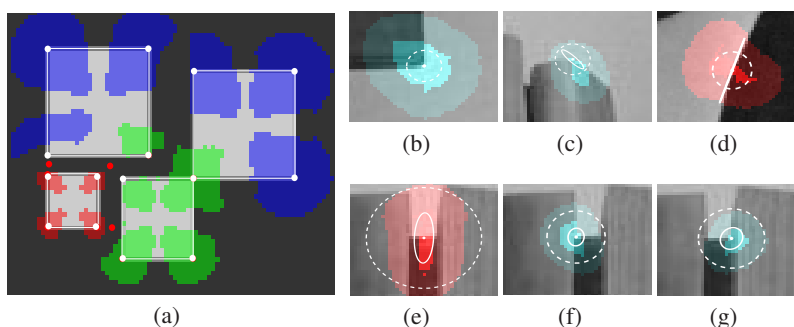


Fig. 6. a) Keypoints and support maps obtained at three different scales. The colors (red, green and blue) indicate the integration scale of the operator which is of 9, 6 and 3 pixels respectively. (b)-(g) Insets of keypoints obtained in real images. The associated uncertainty and RMS errors are depicted as white ellipses and dotted white circles (factor 100x) respectively. The keypoints' support maps, S , (strong color) along with their extensions, S^+ (light color) are superimposed. The red dots indicate a virtual junction.

4.2 Viewpoint Displacement and Repeatability

In this section we examine how viewpoint changes affect the keypoint detection and localization. For this purpose, a series of 81 images are produced after applying homographies from real images to exactly reproduce how this planar scene would be acquired from the positions depicted in Fig.7(a). In the figure, the virtual camera is oriented toward the scene center and moved radially from this point. More precisely, the camera covers a zenith¹ angle varying from -45 to +45 degrees with 5 degree steps and an azimuthal² angle varying from 0 to 135 degrees with 45 degree steps. The detector was tested on multiple scenes among which 3 different cases are presented here.

We compare the ASN detector with the SIFT and Forstner operators. When it is required, the image gradient is computed by the convolution of a Gaussian derivative of $\sigma_d = 1$. The integration scale of the Forstner operator is set to $\sigma_I = 3$. Both, the ASN and Forstner operators reject a keypoint when the eigenvalue ratio is 0.1. We used the SIFT implementation that is available online. For a given image, the ASN and Forstner operators find an equivalent number of keypoints while SIFT typically finds two to three times more keypoints.

Since we know the exact mapping between the images, we can align the points extracted between two images and measure any bias present. In this experiment, the points detected in each image are mapped to the front view image. This reference image is taken at a zenith angle of zero degree. A point is successfully matched if the actual homography between the images maps the detected point within a distance of 0.7 pixel of a point previously transferred to the reference image. This method is similar to tracking and allows a point to be matched over a large baseline while avoiding false matches.

¹ Measured from the Z-axis

² Measured from the X-axis in the XY-plane

We first calculate the mean repeatability of an operator at each zenith angle. The repeatability for a given image is obtained as the ratio of the number of matched points between the image and the reference image, to the number of detected points in the reference image. The mean repeatability for the three operators is plotted with respect to viewpoint in Fig.7. This figure shows equivalent repeatability between ASN and Forstner. The SIFT operator lower score is due to the large number of keypoints detected in each image.

Besides repeatability, we compute the maximum displacement of the points with respect to viewpoint, which indicates the capability of an operator to identify the same physical point. The displacement of a point is the norm of the vector between the point mapped from a given image to the reference image, and its correspondent detected in the reference image. Since no noise is added to the synthetic images, the displacement is caused by the estimation bias such as explained in previous sections. The maximum displacement of a point is computed over all azimuthal angles for a given zenith angle. The histograms in Fig.8 compile the point maximum displacements observed with respect to the zenith angle, from left to right for the ASN, Forstner and SIFT operators respectively. The white and black dotted lines depict the maximum and mean displacement values at a given zenith angle.

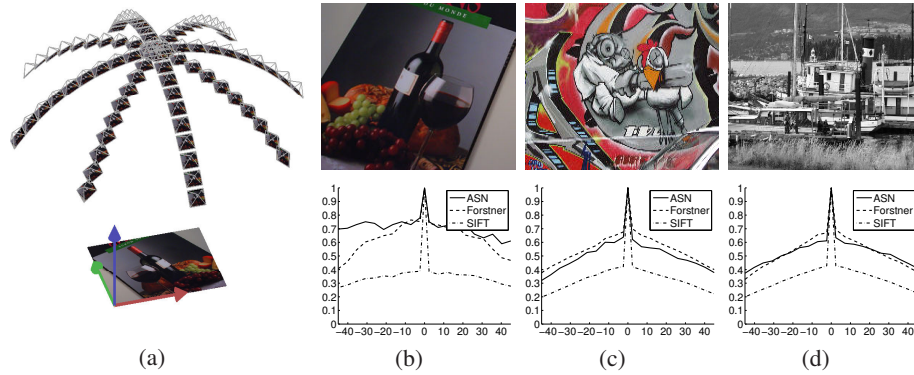


Fig. 7. a) Positions of the virtual camera where the images of the planar scene are acquired. (b)-(c)-(d) Repeatability under viewpoint variation for three different scenes. The results obtained for a given scene are displayed below the reference image. The horizontal axis is the zenith angle of the viewpoint, in degrees.

The highest displacement is observed for SIFT features. To explain such results we refer back to Fig.1(b). The SIFT operator detects region centers based on the Difference of Gaussian (DoG) function. However, SIFT keypoints are also detected near junctions. Actual keypoints arising from these situations are displayed in Fig.1(b). On the right, a stable keypoint is detected over a small region. The DoG function benefits from the rotational symmetry of the underlying structure. On the left, the keypoint is located near a junction in the image and is systematically shifted as the camera moves along the tra-

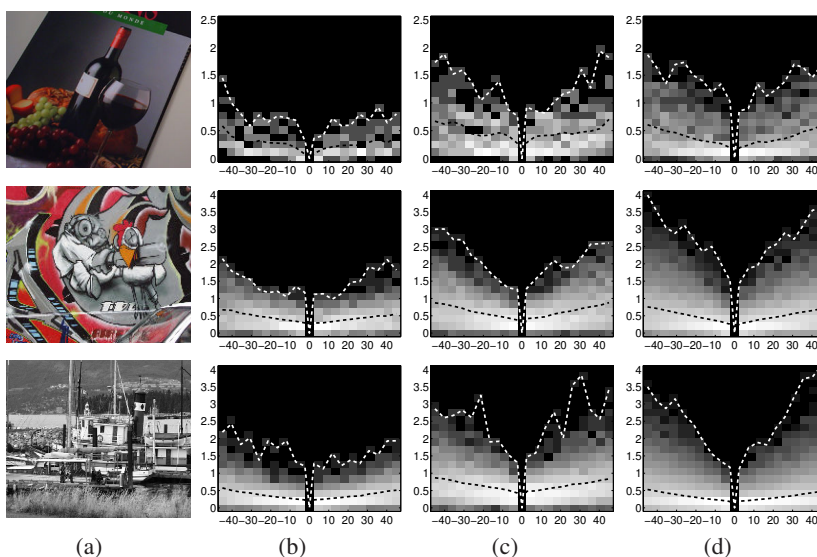


Fig. 8. Histograms of the maximum point displacements under viewpoint variation for three different scenes. The columns present vertical histograms of the maximum displacement (in pixels) with respect to the zenith angle (in degrees) for the ASN (b), Forstner (c) and SIFT (d) operators. In the histograms, the brightness is proportional to the logarithm of the number of points with displacement indicated on the vertical axis. The white and black dotted lines represent the maximum and mean displacement values at a given zenith angle.

jectory. The operator is more sensitive to non-symmetric regions. The latter situation is commonly encountered and explains the high displacement values for SIFT keypoints.

From the histograms in Fig.8, the maximum displacements of the keypoints extracted with the ASN operator are smaller with the viewpoint, regardless of the scene. The maximum displacements originate from image structures of low curvature and the detection of such situations from a single view is still a challenge. Interestingly, both ASN and Forstner operators detect the same number of points and exhibit similar repeatability (see the bottom row of Fig.7). Yet, as it can be observed in Fig.8, systematically for all scenes, the keypoints found by the ASN operator display smaller position variations. The difference arises from the adaptive estimates as well as the absence of Gaussian weight in the estimation window for the ASN detector.

5 Conclusion

The principle of the ASN detector differs fundamentally from other keypoint detectors. At each position in the image, it systematically estimates the position of a potential junction from the local gradient field in the image. Keypoints then appear where multiple position estimates are attracted. These are poles. Since a keypoint results from estimations at several positions of a window neighborhood, the detector adapts to im-

age content and one obtains the area where the keypoint is a local landmark, namely the support map. It was further shown how the ASN detector adapts integration scale for identifying physical points. By prioritizing larger scales and avoiding Gaussian weight in the estimation, improved accuracy and stability with viewpoint change were shown. This type of keypoints will be useful to improve accuracy when combined with less accurate keypoints featuring rich descriptors. In future work, it will also be interesting to investigate the role of the support map for disambiguating correspondences.

References

1. Schmid, C., Mohr, R., Bauckhage, C.: Evaluation of interest point detectors. *International Journal of Computer Vision* **37**(2) (2000) 151–172
2. Lowe, D.G.: Object recognition from local scale-invariant features. In: *International Conference on Computer Vision*. (1999) 1150–1157
3. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* **60**(2) (2004) 91–110
4. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. *International Journal of Computer Vision* **60**(1) (2004) 63–86
5. Bay, H., Tuytelaars, T., Gool, L.V.: Surf: Speeded up robust features. In: *European Conference on Computer Vision*. (2006) 404–417
6. Rohr, K.: Localization properties of direct corner detectors. *Journal of Mathematical Imaging and Vision* **4**(2) (1994) 139–150
7. Deriche, R., Giraudon, G.: Accurate corner detection: an analytical study. *International Conference on Computer Vision* (1990) 66–70
8. Tuytelaars, T., Gool, L.V.: Wide baseline stereo matching based on local, affinely invariant regions. In: *British Machine Vision Conference*. (2000) 412–422
9. Lindeberg, T.: Feature detection with automatic scale selection. *International Journal of Computer Vision* **30**(2) (1998) 79–116
10. Triggs, B.: Detecting keypoints with stable position, orientation, and scale under illumination changes. *European Conference on Computer Vision* (2004) 100–113
11. Matas, J., Chum, O., Martin, U., Pajdla, T.: Robust wide baseline stereo from maximally stable extremal regions. In: *British Machine Vision Conference*. Volume 1. (2002) 384–393
12. Kadir, T., Zisserman, A., Brady, M.: An affine invariant salient region detector. In: *European Conference on Computer Vision*. (2004) 228–241
13. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *Pattern Analysis and Machine Intelligence* **27**(10) (2005) 1615–1630
14. Fraundorfer, F., Bischof, H.: A novel performance evaluation method of local detectors on non-planar scenes. In: *Computer Vision and Pattern Recognition*. Volume 3. (2005) 33–33
15. Moreels, P., Perona, P.: Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision* **73**(3) (2007) 263–284
16. Forstner, W.: A framework for low level feature extraction. In: *European Conference on Computer Vision*. (1994) 383–394
17. Harris, C., Stephens, M.: A combined corner and edge detector. In: *Alvey Vision Conference*. (1988) 147–151
18. Park, S.J., Ahmad, M.B., Rhee, S.H., Han, S.J., Park, J.A.: Image corner detection using radon transform. In: *International Conference on Computational Science and Applications*. (2004) 948–955