Christopher Hunt
Seminar Review
10 March 2016

Speeded-Up Robust Features (SURF)

I. Introduction

SURF (Speeded-Up Robust Features) can be described as a feature extraction and point correlation framework for use with two-dimensional images. Developed by Herbert Bay, PhD at ETH Zurich, SURF interest points are designed to be in-plane rotation invariant, robust to noise, and extremely fast to calculate. The overall framework can be divided into three phases: interest point detection, interest point description, and interest point matching.

II. Interest Point Detection

One of the reasons that the SURF paradigm is so fast to compute is the image space that all computations are executed in. Instead of traditional RGB or grayscale images, SURF interest points are determined in the integral image space, $I_\Sigma(\mathbf{x})$, where the value of any point $\mathbf{x}$ = (x, y) on the integral image is equal to the sum of the intensities of the area of the rectangle formed between $\mathbf{x}$ and the origin in the grayscale image. Succinctly, this can be quantified by the following equation (1):

$$I_\Sigma(\mathbf{x}) = \sum_{i=0}^{x}\sum_{j=0}^{y} I(i,j) \tag{1}$$

While this image space transform is costly to compute, it can be parallelized to decrease overall computation time. Additionally, this upfront cost significantly reduces further processing time. This is because regions of integral images can be efficiently characterized given simply the values at its vertices. This fact allows for efficient detection of blobs, or areas of constant contrast and color.

The interest point detection portion of this framework revolves around the detection of these aforementioned blobs. In previous work, efficient detection of image blobs can be achieved by analyzing the Hessian of the image. As the Hessian (*H*) is a representation of the curvature of the underlying image, maximal values of the determinant of the Hessian represent regions of constant structure (color, contrast, etc.) within the image. This is the fundamental principal behind Hessian-Laplace detectors. Interest points are defined to be these local maxima of the determinant of the Hessian of the image. Generally, before this computation, the original image is actually convolved with a second-order Gaussian derivative of some standard deviation in order to illicit the characteristic blurring effect that reduces image detail. This convolution reduces noise in the image by smoothing edges and softening colors.

While the Hessian-based detection is an efficient way to produce regions of interest, it is computationally expensive and, therefore, ill-suited for online use. In order to compensate for this, the SURF framework replaces these second-order Gaussian filters with symmetric box filters. By convolving the original image with these x-, y-, and xy-direction box filters and, afterwards, taking the partial derivatives, it possible to cheaply compute an approximate Hessian matrix. As these matrices are simply approximations of one another, the energy of their Gaussian kernels is not conserved. As such, the determinant of the Hessian must have a weighting factor applied. This results in the following equation (2):

$$det(H_{approx}) = D_{xx}D_{yy} - (wD_{xy})^2 \quad (2)$$

D_{xx}, D_{yy}, and D_{xy} refer to the filter responses of the image with the x-, y-, and xy-direction box filters respectively. In order to preserve the Gaussian kernel energy, the weighting factor, *w*, was computed to be 0.9.

In order to match these image regions across different scales, a pyramidal scale space is constructed. This scale space simulates viewing the image from different distances away. Traditionally, this scale space is built iteratively by applying a Gaussian filter, downsampling that image, and repeating this process until a limiting image size is reached. However, this methodology is slow because the process is serial in nature, with each successive level of the pyramid being dependent on its previous level. Instead, the SURF framework proposes an alternative: parallel upscaling. Because the Gaussian filters can be sufficiently approximated with symmetric box filters, it is possible to represent any scale (level of the pyramid) as the convolution of the original image with a filter of size N x N pixels. Because these filters can be made arbitrarily large (smallest possible size is 9 x 9), these filters are able to represent distances arbitrarily far away. Additionally, due to the filters' symmetry, propagating the set of convolution filters is fast. As such, the pyramidal scale space can be constructed in parallel, applying a set of filters to the original image where the dimension of the filter determines its level in the pyramid (9 x 9 is bottom level).

Once regions can be matched through image and scale space, the localization becomes quite easy. The SURF framework uses a non-maxima suppression of a 3 x 3 x 3 neighborhood. Essentially, the gradient of the determinant of the approximated Hessian is followed and, any region of the image that is not considered a local maximum is set to zero. At the end of this process, the regions of the image with nonzero values are determined to be interest points.

III. Interest Point Description

Once interest points can be localized, the SURF framework funnels these points to its descriptor. The interest point description portion of the framework is a two-step process: orientation assignment and feature extraction. It is in this section where the SURF framework really provides novelty, implementing a feature description framework fundamentally different from current viable alternatives, including SIFT and GLOH.

Because the framework aims to produce features which are in-plane rotation invariant, the descriptor requires a way to normalize the point orientations. To this effect, the descriptor assigns a dominant orientation to each interest point using the horizontal and vertical Haar wavelet response. Essentially, the horizontal and vertical Haar wavelets are convolved with a circular neighborhood around an interest point. This will generate horizontal and vertical Haar wavelet responses for each pixel, d_x and d_y respectively. These responses are smoothed for noise reduction and, afterwards, these responses are plotted versus one another. On this plot, each pixel of the original interest point will have xy-coordinate (d_x, d_y).

Once this response plot is generated, a sliding window of size π/3 is created. Within the sliding window, a local orientation vector is generated through the summation of all x- and y-

coordinates encapsulated in the window. That is to say, the local orientation vector ***o*** can be described by the following equation (3):

$$\overrightarrow{o} = \left[\sum_{i=0}^{n} d_x \sum_{i=0}^{n} d_y\right]^T \quad (3)$$

The dominant orientation vector for the interest point then is simply the largest of all such vectors across all such windows.

Once orientation is determined, normalized features can be extracted from the interest point. In order to do this, and object-aligned square neighborhood around the interest point is first defined. This neighborhood's size is dependent on the scale of image, 20*s*. Once a neighborhood is defined, this neighborhood is subdivided into a 4 x 4 grid. An axis-aligned Haar wavelet response (both horizontal and vertical) are both computed against the region in each subdivision. From each subdivision, twenty-five equally spaced points are subsampled to characterize the that region. The twenty-five points are used to compute a local feature vector composed of four features each. This local vector can be seen illustrated in equation (4):

$$\overrightarrow{v} = \left[\Sigma d_x, \Sigma d_y, \Sigma|d_x|, \Sigma|d_y|\right]^T \quad (4)$$

The pure summation features serve as a metric for maximal intensity in either direction while the absolute value summation features serve as a metric for pattern polarity. By concatenating these local feature vectors generated from each subdivision, one can produce the comprehensive, 64-element feature vector that describes the interest point and surrounding neighborhood.

IV. Interest Point Matching

Once interest points have been localized and characterized, the SURF framework begins to build a correspondence between interest points in each image frame. To do this, the framework simply computes a nearest neighbor search in the feature space. As possible quantifiers for "nearest", the authors suggest either Euclidean or Mahalanobis distances. For this particular implementation, sensor precision was assumed to be uniform and, as such, Euclidean distance was sufficient.

In order to efficiently parse the spatial data structure containing the interest points, the SURF framework utilizes a clever trick: Laplacian indexing. In integral image space, the trace of a Hessian matrix, or the Laplace, provides meaningful information on the blob-to-background relationship in the underlying neighborhood. More specifically, the sign of the Laplace is an indicator of the relative brightnesses between the blob and the background. A positive Laplace indicates a bright blob on a dark background, while a negative Laplace indicates a dark blob on a bright background. Given this information, it is possible to use the Laplace of the interest blob as an intuitive hyperplane to divide a spatial data structure (e.g. KD Tree) along. This allows for more efficient elimination of obviously erroneous matching candidates in the new frame. Additionally, this indexing comes at no extra cost as the Hessian is already computed earlier in the detection phase in order to localize these interest points.

V. Validation Results

In order to validate the efficacy of this framework, the authors chose to quantify two important metrics: the repeatability of the detector and the discriminative power of the descriptor. The repeatability was defined as the percentage of interest points originally localized in one camera viewpoint that were again localized in another camera viewpoint. The discriminative power was defined as the detection-false positive relationship generated when using a simple bag-of-words classifier. In effect, both of these metrics are measures of how robust the features generated by the SURF framework are. The testing scenarios and results will be discussed at length below.

In order to quantify the repeatability of the detector, interest points were generated on four different sets of images: Graffiti, Wall, Bikes, Boats. Each set of images was essentially a series of snapshots of a single object from different angle viewpoints. This angle only varied around a single axis and, as such, it was possible to map the repeatability metric to an angle offset. In this case, they considered the 0º offset to be the ground truth and compared the interest points generated in every other image to those generated at 0º. Repeatability results were generated for the SURF's Fast Hessian 9x9 variant (FH-9) as well as SURF's Fast Hessian 15x15 variant (FH-15). Along with those, a Difference of Gaussians (DoG), Hessian-Laplace, and Harris-Laplace detector were all tested as well. In all the given testing scenarios, the FH-15 detector performed the best with ~78% maximal repeatability at 20º offset. Competitively following slightly behind, the FH-9 detector also reaches a maximal repeatability of ~75 % at 20º offset. While repeatability decreased as the offset angle increases, which is to be expected, the repeatability percentage for both FH detectors remained greater than 50 % up to a 50º offset. This is markedly better than the behavior of its competitors.

In order to quantify the discriminative power of the descriptor, the features generated by the descriptor would be used as features for a publicly available bag-of-words classifier. Using a set of four hundred images, supervised training of the classifier was performed using the first two hundred images and testing was performed on the second two hundred. The intuition behind this experimental set up is that the more characteristic the feature vector produced by the descriptor is, the more accurate the classifier should be. This accuracy can be understood as the ratio between the number of false positives and the total number of detections. For this test, the SURF-128 descriptor was compared against SIFT and GLOH. According to these results, the SURF-128 descriptor's features are much more characteristic of the underlying image space. This is evident by the near-ideal shape of the produced classification curve. For SURF-128 , the false alarm probability didn't go above 0.1 until the detection probability reached ~0.85. This is a leaps and bounds better than the SIFT or GLOH results, ~0.55 and ~0.5 respectively. Of course, this is using interest points generated from the SURF detector and so there is an inherent bias. When using simply random edge pixels, the SURF-128 performs only marginally better than SIFT, however, there is still a large increase in performance when compared to GLOH.

VI. Opinion and Conclusion

In my opinion, this paper does many things very well but still has areas it could have improved upon. Of the things it does well, I believe the most important is that the narrative is, for the most part, self-contained. It is completely possible to follow the logic of the SURF- framework without searching deeply through its references. Another thing that is done well is the inclusion of an applications portion to the original article. This portion exemplifies how SURF can be used in

common computer vision tasks, such as 3D scene reconstruction or object recognition, and adds much needed context to the framework. Additionally, this section serves to illustrate the immediate impact the framework can have on the field. Finally, I appreciate the honesty of the article. In its introduction, the framework professes robust and speedy feature calculations and that is exactly what is delivered.

Of the things that can be improved with this paper, I believe the most important to be a thorough exploration of the speed metrics. The hallmark of SURF is its speed and, unfortunately, in this paper, the speed seems to be abstracted into relativities. SURF is constantly claimed to perform faster than various other frameworks, however, no quantitative assessment of these statements are given. This seems to be an inexact and vague qualitative assessment instead. Additionally, the validation experiments for the descriptor seems shallow. They only used a single data set (e.g. Caltech Airplanes and Backgrounds) to validate the efficacy of the classifier and I believe that that is limiting. It would have been preferable to perform the same experiment over three or four data sets, much like the detector validation. For future work, it would be interesting to see the SURF framework applied to very different data sets.

VII. References

Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. "SURF: Speeded Up Robust Features." Computer Vision – ECCV 2006 Lecture Notes in Computer Science (2006): 404-17. Web.