Evaluating Performance of Feature Extraction Methods for Practical 3D Imaging Systems

Deepak Dwarakanath[1,2], Alexander Eichhorn[1], Pål Halvorsen[1,2], Carsten Griwodz[1,2]
[1]Simula Research Laboratory, [2]University of Oslo
Oslo, Norway
{deepakd, echa, paalh, griff}@simula.no

ABSTRACT

Smart cameras are extensively used for multi-view capture and 3D rendering applications. To achieve high quality, such applications are required to estimate accurate position and orientation of the cameras (called as *camera calibration-pose estimation*). Traditional techniques that use checkerboard or special markers, are impractical in larger spaces. Hence, feature-based calibration (*auto-calibration*), is necessary. Such calibration methods are carried out based on features extracted and matched between stereo pairs or multiple cameras.

Well known feature extraction methods such as SIFT (Scale Invariant Feature Transform), SURF (Speeded-Up Robust Features) and ORB (Oriented FAST and Rotated BRIEF) have been used for auto-calibration. The accuracy of auto-calibration is sensitive to the accuracy of features extracted and matched between a stereo pair or multiple cameras. In practical imaging systems, we encounter several issues such as blur, lens distortion and thermal noise that affect the accuracy of feature detectors.

In our study, we investigate the behaviour of SIFT, SURF and ORB through simulations of practical issues and evaluate their performance targeting 3D reconstruction (based on epipolar geometry of a stereo pair). Our experiments are carried out on two real-world stereo image datasets of various resolutions. Our experimental results show significant performance differences between feature extractors' performance in terms of accuracy, execution time and robustness to blur, lens distortion and thermal noise of various levels. Eventually, our study identifies suitable operating ranges that helps other researchers and developers of practical imaging solutions.

Categories and Subject Descriptors

I.4 [**Image Processing and Computer Vision**]: Digitization and Image Capture-Camera calibration, Imaging geometry; I.4 [**Image Processing and Computer Vision**]: Segmentation-Edge and feature detection

IVCNZ'12, November 26 - 28 2012, Dunedin, New Zealand

Keywords

Feature detection & extraction, Stereo matching performance, SIFT, SURF, ORB, Auto calibration, 3D applications.

1. INTRODUCTION

Multi-view vision applications such as free-view rendering [3], motion tracking [11], structure from motion [16], and 3D scene reconstruction [10] require precise geometrical information about location and pose of each camera. Traditionally, camera calibration techniques use checkerboards [19, 18] or special markers [5] to find point correspondences between images. While such methods achieve sufficient accuracy they are often inconvenient and limited in practice. In some cases, it is impossible to place a measurement target like a checkerboard pattern of sufficient size in a scene. Automatically finding corresponding points based on image features alone is thus a desired goal.

To avoid dedicated calibration patterns and special markers in a scene, several auto-calibration methods have been proposed [6] [7]. They rely on matching automatically detected feature points between images from different camera perspectives to estimate geometrical calibration parameters. Feature extractors like SIFT (Scale Invariant Feature Transform) [8], SURF (Speeded-Up Robust Features) [1] and ORB (Oriented FAST and Rotated BRIEF) [15] are widely used due to their easy availability, good detection and matching performance, and a relatively small computational cost. However, little is known about their spatial accuracy and robustness to real-world distortion although these issues play a major role for precise reconstruction and scene geometry.

Sun et al. [17] have shown that the accuracy of calibration is sensitive to the quality of corresponding features. At least 7 matching points are required for an accurate estimation of calibration parameters [4] and more points will usually improve the performance.

Most feature extraction algorithms are optimized for image recognition tasks and search scenarios instead of geometry calibration. Hence, typical performance metrics such as repeatability, precision and recall or number of matches only consider the performance of matching [12], rather than the performance in terms of 3D geometry reconstruction.

Moreover, the quality of images obtained from real-world imaging sensors suffers from practical issues such as defocus and motion blur, different lens distortions, thermal noise, offsets in exposure time and white balance. Such perturbations may degrade the performance of feature extraction and matching up to a point where geometry reconstruction accuracy becomes unacceptable.

Our study provides practical insights about the robustness of existing feature extractors obtained in real-world experiments and simulations. We seek to understand the typical operation ranges of three prominent feature extraction methods; SIFT, SURF and ORB. We particularly investigate how different image distortions can impact the precision of camera pose estimation when relying on detected and matched feature points. We evaluate 3D calibration performance based on extracted image features under different levels of quality degradation. We use two real-world video data sets with a medium depth range, both captured in-doors from multiple camera perspectives. We simulate image quality degradation by introducing several levels of gaussian blur, geometrical lens distortion and sensor noise. To measure the geometrical accuracy of feature-based calibration, we use a performance metric derived from the epipolar constraint [4] which defines a precise geometrical relation between a stereo pair of images. Together with an analysis of computational costs we identify suitable operation ranges to aid researchers and developers of multi-view applications.

In our experiments, we find substantial differences in robustness and execution time between SIFT, SURF and ORB. SIFT and SURF are more robust, than ORB, to defocus, lens distortion and thermal noise. Although SURF performs similar to SIFT in terms of accuracy, SURF reduces the computational cost drastically, by almost half. Comparatively, ORB is the most computationally efficient extractor at higher resolutions and is robust to lens distortion, but accuracy is inadequate for defocused and noisy images.

2. FEATURE EXTRACTORS

In this section, we briefly explain the principle of operation of SIFT, SURF and ORB feature extractors.

SIFT detects key points in an image that are highly distinct, scale and rotation invariant, and fairly invariant to illumination. SIFT is computed as follows. First, the interesting points are searched over scale-space representation of a image, and a difference of the Gaussian function is used to identify the interesting points, which are invariant to scale and orientation. The interesting points are subjected to a 3D quadratic function to determine their location and scale. Every key-point is assigned one or more orientations depending on the direction of local gradients of the image around this key-point and a highly distinct 128-bit descriptor is computed.

SURF uses novel schemes for detection and description, which mainly focuses on reducing computational time. Integral images are computed and interesting points are obtained based on the Hessian matrix approximation. Using scale-space representation, interesting points are searched over several scales and levels. Localization: carried out using interpolation of space. This is important because number of interesting points in different layers of scales are large. The descriptor is built using the distribution of intensity content within the interesting points. SURF uses distribution of first order Haar wavelet responses in, both the x and y directions. An additional step of indexing is based on the sign of the Laplacian to increase robustness and matching speed.

ORB modifies the FAST [14] detector to detect key points by adding a fast and accurate orientation component, and uses the rotated BRIEF [2] descriptor. Corner detection using FAST is carried out and that results in N points that are sorted based on the Harris measure. A pyramid of the image is constructed, and key points are detected on every level of the pyramid. Detected corner intensity is assumed to have an offset from its center. This offset representation, as a vector, is used to compute orientation. Images are smoothened with the 31 x 31 pixel patch. Orientation of each pixel patch is then used to steer the BRIEF descriptor to obtain rotational invariance.

3. EVALUATION OVERVIEW

3.1 Simulation parameters

All imaging systems encounter practical issues such as defocus, radial lens distortion and thermal noise. *Image blur* is the loss of image sharpness caused due to defocus, shallow depth of field and motion of the camera or the scene objects and quantization process. In our study, we focus on image blur due to defocus only, because we consider multi view capture using only stationary cameras and hence motion blur is of lesser significance. *Radial lens distortion* is an optical aberration caused by spherical lens surfaces of the cameras, which produces aberrations symmetrically and radially from the image center. Barrel and pincushion are the types of radial distortions where the image aberration increases and decreases respectively as the radial distance from image center increases. *Image thermal noise* appears as random speckles in an image which is random variation in the luminosity or color information of the pixels caused by the camera sensor and its circuitry. To study the performance of the feature extractors under such practical scenarios, we simulate defocus, lens distortion and noise using the mathematical models.

Defocus $I_b(u,v)$ is accomplished by smoothing an image $I(u,v)$ with a linear 2D Gaussian filter $G(u,v)$, as in equation 1. Various defocus levels can be controlled by the variance σ_b of the Gaussian kernel, which represents blur radius.

$$I_b(u,v) = I(u,v) * G(u,v) \tag{1}$$

$$G(u,v) = \frac{1}{2\pi\sigma_b^2} e^{-\frac{u^2+v^2}{2\sigma_b^2}} \tag{2}$$

Lens distortion can be modeled as a 3rd order polynomial, as given by equation 3, where R_u and R_d is undistorted and distorted pixel radius, respectively. The distortion co-efficient k_1 can be varied to obtain various levels of distortion.

$$R_u = R_d + k_1 R_d^3 \tag{3}$$

Thermal noise is modeled as Gaussian distribution. A noisy image $I_n(u,v)$ is obtained by adding Gaussian random noise $N(u,v)$ with zero mean and variance σ_n to an image $I(u,v)$, as in equation 4. To obtain various noise levels N_l, measured in decibels, the variance σ_n is controlled as, $\sigma_n = 10^{N_l/10}$.

$$I_n(u,v) = I(u,v) + N(u,v) \tag{4}$$

3.2 Performance measure

The performance of feature extractors are measured in terms of accuracy, detectability and execution time.

Accuracy of feature extraction in stereo images is measured by deviations of measured positions of matched feature points from their ideal positions. To explain this in detail, we bring in the concept of *epipolar geometry*. Researchers [4] [9] [13] have shown that in 3D imaging systems,

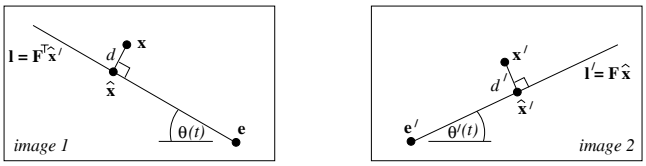


Figure 1: Illustration of Epipolar Geometry. Courtesy R. I. Hartley [4]

the geometrical relationship between the point correspondences between stereo images is important and is characterized by a mapping matrix called *Fundamental Matrix* (F).

The epipolar geometry is illustrated in figure 1. Ideally, for every point in one of the stereo images (say $\hat{x}$), a corresponding point on the other stereo image ($\hat{x'}$) should lie on a line, called *epipolar line* (l'), which is computed using the matrix (F). In practice, feature extractors estimate the corresponding point (x'), which can lie outside the line and thus producing an error (d'). Such an error averaged over all N_p feature points will be referred as *Epipolar Error* (E_p), and can be computed as in equation 5. Thus, the *Epipolar Error* aids in measuring the accuracy of feature extractors, in pixels. The sub-pixel errors, that is $Ep < 1$ pixel, is an acceptable value for good performance in most of the relevant applications.

$$E_p = \sum_{i=1}^{N_p} \frac{x_i' F x_i}{(Fx_i)_1^2 + (Fx_i)_2^2 + (F^T x\prime_i)_1^2 + (F^T x\prime_i)_2^2} \qquad (5)$$

Detectability measures the ability to obtain sufficient feature point correspondences in stereo images. A good estimation of Fundamental Matrix requires at least 7 feature corresponding points in stereo images [4]. Therefore, the percentage of trials resulting in at least 7 feature correspondences represents the detectability of a feature extractor.

Execution time measures the computational speed of the feature extractors. It is computed as time spent on the extraction step (detecting interesting points in two images and building descriptors for them) and the matching step (performing feature matching to obtain feature correspondences).

3.3 Simulation Setup

Our experimental setup, as illustrated in figure 2, comprises a database of the test stereo images, an image degradation module and a feature extraction and matching module. During our evaluation, stereo images are retrieved from the database, and the image degradation module pre-transforms the stereo images to simulate defocus, lens distortion and sensor noise, with various levels using a tuner. Then, the feature detector-descriptor-matcher operates over all stereo images that are pre-transformed. The resulting feature matches on degraded images are used to evaluate the performance of the feature extractor based on the fundamental matrix estimated for the stereo images before degradation.

In our experiments, we have used 30 stereo images from the dataset of an opera performance, captured using 8 cameras (2 camera arrays, each consisting of 4 cameras of narrow and wide angle lens respectively). A second dataset used for evaluation contains 35 images, from the popular breakdance

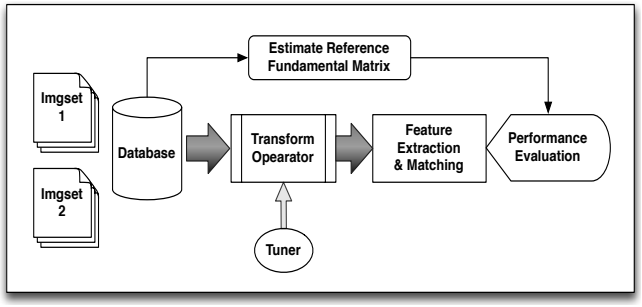


Figure 2: Evaluation Pipeline

video sequence from Microsoft [20]. The stereo images from both datasets were of HD resolution (1280x768). All these stereo images were scaled to high resolution (1280x960), medium resolution (640x480) and low resolution (160x120) images to study the behavior of feature extractors across various resolutions in conjunction to image degradation. Image degradation was carried out at different levels on every test stereo pair (equally on both images of a stereo pair). Blur radius levels ranged from values 1.5-6.0. Barrel distortion and pincushion distortion were varied as -50% to -10% and +10% to +50% respectively. Thermal noise levels were 5 - 50dB. Then, feature extraction and matching using SIFT, SURF and ORB methods are performed and the performance is evaluated. An example of feature extraction in stereo images for various datasets and the image degradation using simulation parameters are shown in figure 3.

4. EVALUATION RESULTS

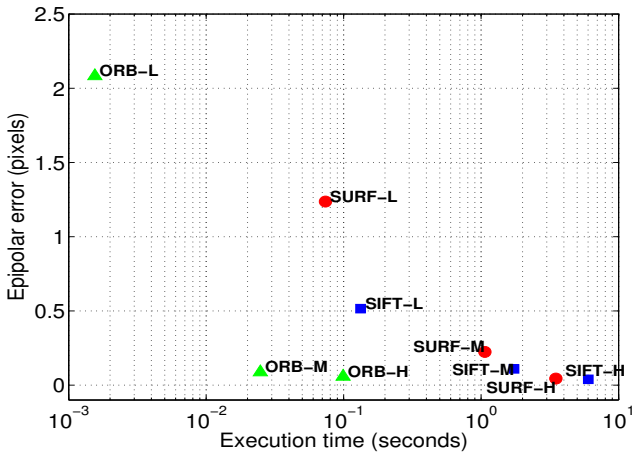
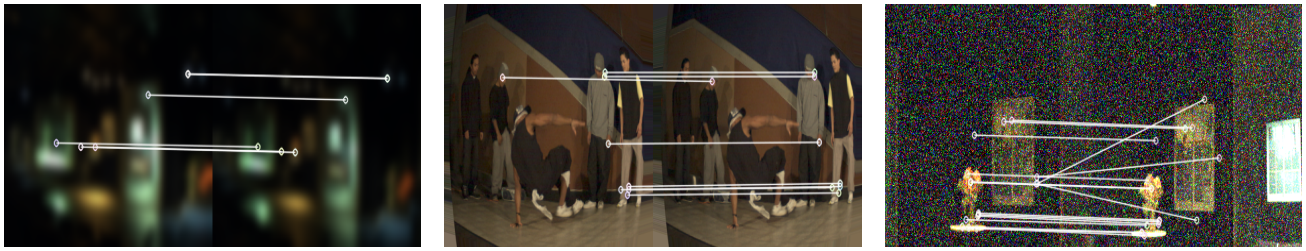


Figure 4: Accuracy Vs Computational time. L- low (160x120) resolution, M- medium resolution (640x480), H- high resolution (1280x960)

First, we ran the tests to measure accuracy and execution time of various feature extractors to comparatively analyze the performance of feature extractors at various image resolutions. Figure 4 shows the results of the test (note that the execution time is plotted in logarithmic scale). Obviously, a tradeoff exists in choosing feature extractors between achiev-

(a) Opera dataset - Wide lens with SIFT points (b) Microsoft dataset with ORB points (c) Opera dataset - Narrow lens with SURF points



(d) SIFT matching on blurred images (e) ORB matching on distorted images (f) SURF matching on noisy images

Figure 3: Stereo images from various datasets of resolution 320x240

ing higher accuracy and higher speed. Overall, ORB is computationally efficient compared to SIFT and SURF at all resolutions. A relative difference in execution time between SIFT and SURF is significant; SURF reduces the computational cost by 48% at all resolutions. SIFT, SURF and ORB results in acceptable (sub-pixel) accuracies, except for SURF-L and ORB-L. This shows that SIFT is more robust to change in scale.

Next, we conducted experiment to discuss how the defocus/image blur, lens distortion and thermal noise affects the performance of feature extractors, and the results are shown in figure 5.

4.1 Effects of blur variation

Figures 5(a), 5(b) and 5(c) show that SIFT outperforms in terms of accuracy at all resolutions. SIFT seems to be robust to blur levels probably because of its own way of finding key points, which uses scale space representation with various blur levels. SIFT operations on blurry images are equivalent to having more levels of blurs in every octave of the scale space, for an un-blurry image. Obviously, at lower resolutions blur-ness has a greater effect and hence SIFT shows an acceptable accuracy up to blur level 4.5, as in figure 5(c).

SURF performs marginally at acceptable accuracy ($E_p <= 1$, figure 5(c)) up to blur level 4.5, at low resolutions, for the same reasons mentioned for SIFT. However, the difference in accuracies between SURF and SIFT is due to the descriptor construction. SURF integrates the gradient information and loses distinctiveness when blur increases, while SIFT uses individual gradient to create the descriptor and sustains the performance to a larger extent of blur, compared to SURF.

The detectability measure (figure 5(d)) for both SIFT and SURF reduced drastically with increase in blur level at low resolution, which makes them unsuitable to use when low resolution images are blurred, especially at levels > 4.5.

Although, ORB performs good only at medium and high resolution (figures 5(b) and 5(a)) up to blur level 3.5, the detectability of ORB decreases rapidly with increase in blur level. The use of huge box filters in ORB to obtain descriptors seems to limit performance on blurry images. Additional blur worsens the efficiency of the descriptor. Hence ORB fails at low resolutions.

4.2 Effects of distortion variation

The effects on performance of the feature extractors due to different levels of barrel and pincushion distortion can be seen in figures 5(e), 5(f), 5(g) and 5(h). All the feature extractors perform well and similar at high and medium resolution. A at low resolutions, SIFT outperforms SURF, which in turn outperforms ORB; however, all of them exhibit an acceptable accuracy and a constant detectabitlity. Overall performance of SIFT, SURF and ORB at all resolution andseems to be unaffected by lens distortion. It should be noted that this result is for a homogenous stereo pair where the distortions are assumed to be of same degree in both the cameras.

4.3 Effects of noise variation

The measurements for this experiment peaked at around 10 pixels, hence the results are shown in log scale for y axis in figures 5(i), 5(j) and 5(k). Here, we show that SIFT outperforms SURF and ORB, at all resolutions and exhibited resilience to thermal noise, but becomes sensitive to noise at around 15dB for low resolution images. SURF and ORB showed resilience to noise up to 20dB and 15dB, respectively, at both high and medium resolutions. Importantly, we observe high and constant detectability rate (figure 5(l)) for SURF and ORB, suggesting that the performance of SURF and ORB are not affected by noise, but the accuracy is too low ($E_p > 10$ pixels). This behavior is because SURF and ORB detect more features which are not supposed to be, in noisy images. Hence under noisy conditions, above 15dB none of the feature extractors perform within the acceptable accuracy.

5. CONCLUSION

In this paper, we evaluated the popular and widely used feature extractors SIFT, SURF and ORB. The experiments were conducted over different datasets at various resolutions to test the resiliency of the feature extractors to defocus/blur, lens distortion and thermal noise. From the results, we can conclude that:

- At resolutions > 320x240, SIFT and SURF are the best choices. However, choosing SURF would save execution time of 48%, on an average, with a cost of around 0.10 pixels in accuracy. A choice of feature extractor should be made considering the below conclusions, which are based on the resolution 320x240.
- For blurry images, SIFT is the best choice. However, using SURF would save 48%, on an average with a cost of 0.22 pixels in accuracy.
- For lens distorted images, SIFT, SURF and ORB all are good choices. By using ORB, the execution time reduces by 98.12% and 95.27% with a cost of 0.69 pixels and 0.33 pixels in accuracy compared to SIFT and SURF, respectively.
- For noisy images, SIFT and SURF are good choice and using SURF saves 32% time with a cost of 0.67 pixels in accuracy.

Unlike other feature evaluations, we have used the Epipolar Error to measure the accuracy of the feature correspondence, which aids to selection of feature extractors for feature based calibration and other 3D applications.

6. REFERENCES

[1] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer Vision and Image Understanding*, 110:346–359, June 2008.

[2] M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. In *European Conference on Computer Vision*, September 2010.

[3] M. Dongbo et al. 2d/3d freeview video generation for 3dtv system. In *Proceedings of the IEEE International Conference on Image Processing*, pages 1760 –1763, October 2008.

[4] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[5] G. Kurillo et al. Wide-area external multi-camera calibration using vision graphs and virtual calibration object. In *Proceedings of the ACM/IEEE International Conference on Distributed Smart Cameras*, pages 1 –9, September 2008.

[6] C. Li et al. A camera on-line recalibration framework using sift. *The Visual Computer: International Journal of Computer Graphics*, 26:227–240, March 2010.

[7] R. Liu et al. Stereo cameras self-calibration based on sift. *International Conference on Measuring Technology and Mechatronics Automation*, 1:352–355, 2009.

[8] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, November 2004.

[9] Y. Ma et al. *An Invitation to 3D Vision: From Images to Geometric Models*. Springer Verlag, 2003.

[10] W. Matusik et al. Image-based visual hulls. In *Proceedings of the Conference on Computer Graphics and Interactive Techniques*, pages 369–374, 2000.

[11] T. B. Moeslund and E. Granum. A survey of computer vision-based human motion capture. *Computer Vision and Image Understanding*, 81:231–268, March 2001.

[12] P. Moreels and P. Perona. Evaluation of features detectors and descriptors based on 3d objects. *International Journal of Computer Vision*, 73:263–284, July 2007.

[13] F. Olivier. Three-Dimensional Computer Vision – A Geometric Viewpoint. *MIT Press*, 1996.

[14] E. Rosten and T. Drummond. Machine learning for high-speed corner detection. In *European Conference on Computer Vision*, volume 1, pages 430–443, May 2006.

[15] E. Rublee at al. Orb: An efficient alternative to sift or surf. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 2564 –2571, nov. 2011.

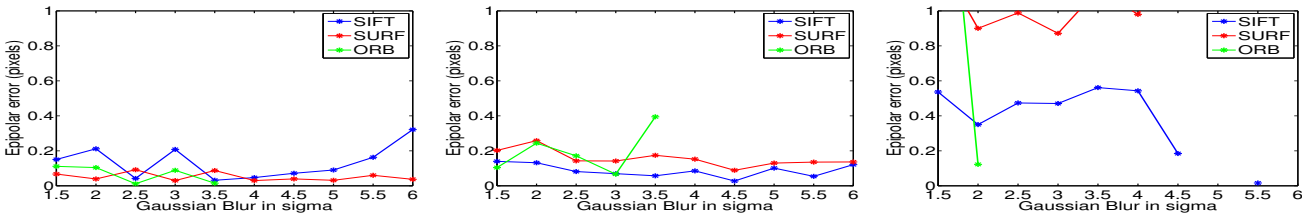
[16] A. Sameer et al. Building rome in a day. In *International Conference on Computer Vision*, pages 72–79, 2009.

[17] W. Sun and J. Cooperstock. An empirical evaluation of factors influencing camera calibration accuracy using three publicly available techniques. *Machine Vision and Applications*, 17:51–67, 2006.

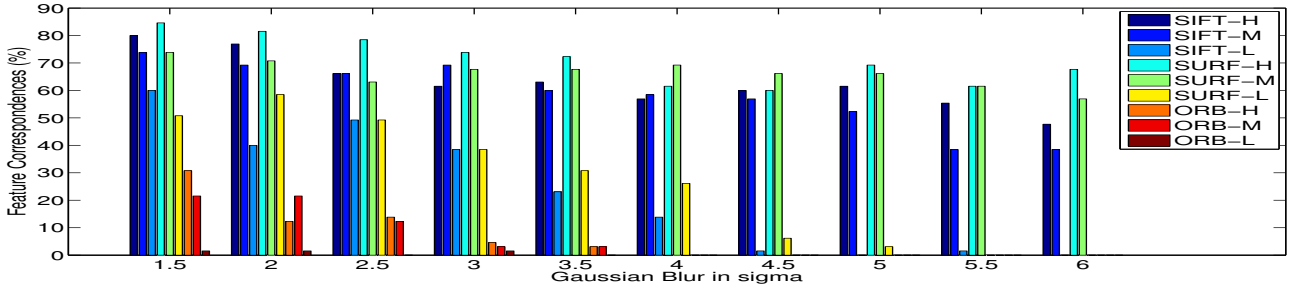
[18] R. Tsai. A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses. *IEEE Journal of Robotics and Automation*, 3(4):323 –344, August 1987.

[19] Z. Zhang. A flexible new technique for camera calibration. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(11):1330 – 1334, November 2000.

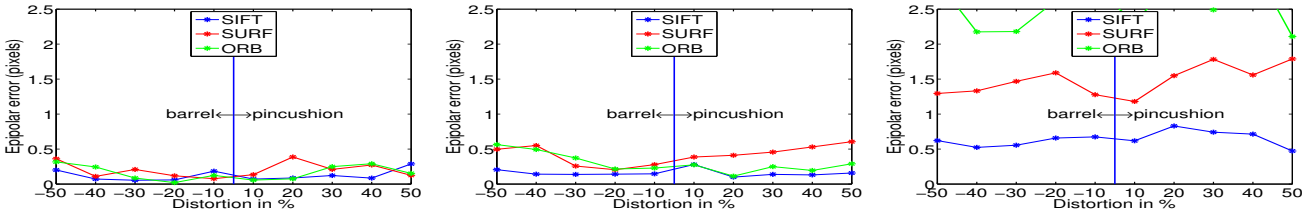
[20] C. Zitnick et al. High-quality video view interpolation using a layered representation. *ACM SIGGRAPH and ACM Transactions on Graphics, Los Angeles, CA*, pages 600–608, August 2004.



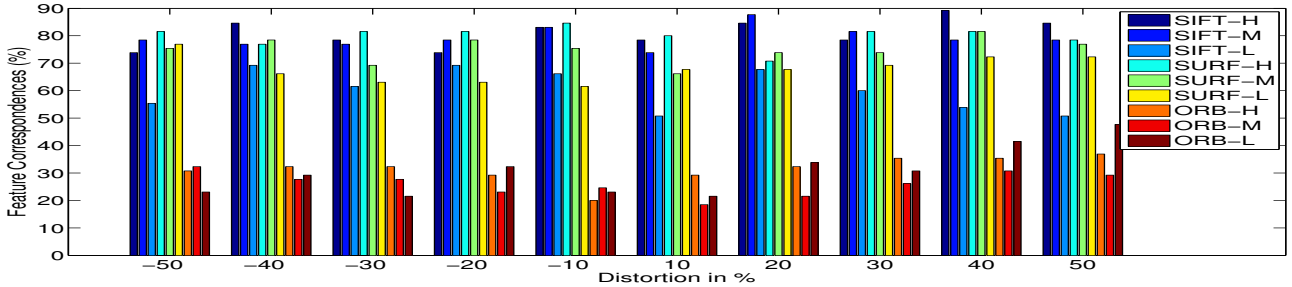
(a) Blur at High Resolution (b) Blur at Medium Resolution (c) Blur at Low Resolution



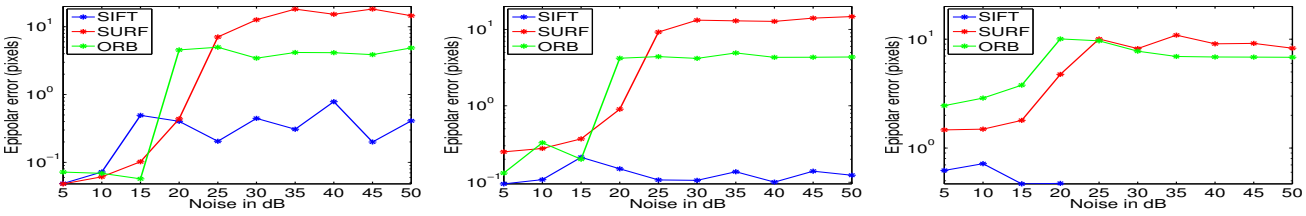
(d) Detectability for blurred images



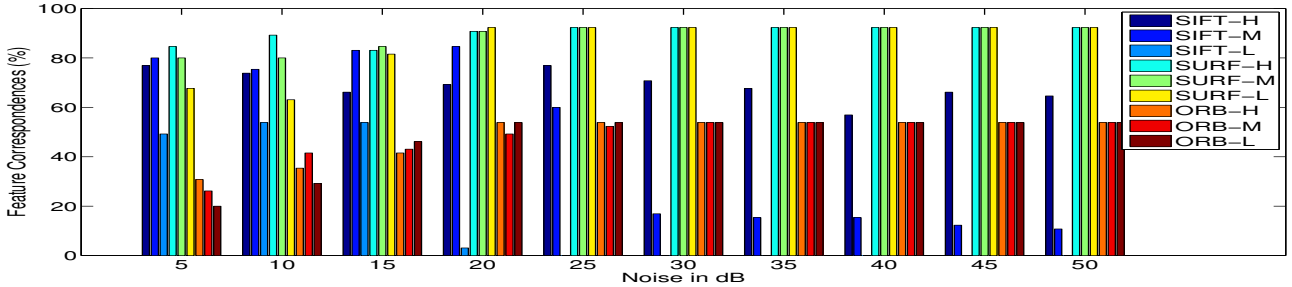
(e) Distortion at High Resolution (f) Distortion at Medium Resolution (g) Distortion at Low Resolution



(h) Detectability for Distorted images



(i) Noise at High Resolution (j) Noise at Medium Resolution (k) Noise at Low Resolution



(l) Detectability for Noisy images

Figure 5: Performance of feature extractors for simulation of blur, distortion and noise levels over various resolutions