Salient Keypoint Selection for Object Representation

Paper ID: 1570232318
Twenty Second National Conference on Communications : NCC 2016

Authors: Prerana Mukherjee, Siddharth Srivastava, Brejesh Lall

Department of Electrical Engineering
Indian Institute of Technology, Delhi

OVERVIEW

Salient Keypoint Selection for Object Representation

- **Introduction**
- **Background**
- **Proposed Methodology**
- **Experimental Results and Discussions**
- **Conclusion**

INTRODUCTION

- We propose a keypoint selection technique which utilizes SIFT and KAZE keypoint detectors, a texture map and Gabor Filter.
- The obtained keypoints are a subset of SIFT and KAZE keypoints on the original image as well as the texture map.
- These are ranked according to the proposed saliency score based on three criteria:
  - distinctivity,
  - detectability
  - repeatability
- These keypoints are shown to be effectively able to characterize objects in an image.

INTRODUCTION

- Selecting relevant keypoints from a set of detected keypoints assists in reducing:
  - the computational complexity
  - error propagated due to irrelevant keypoints.

- This would help in application domains where objects are primary concern such as object classification, detection, segmentation etc.

Motivation

Most matchable keypoints: regions with reasonably high Difference of Gaussian (DoG) responses. [1]

KAZE features have strong response along the boundary of objects while SIFT captures shape, texture etc. similar to neuronal response of human vision system. [6]

KEY CONTRIBUTIONS

- **First work using KAZE with SIFT keypoints for keypoint selection aimed at object characterization and its subsequent use for object matching.**

- **Salient Keypoint selection of SIFT features on Gabor convolved image for representation of features inside object boundaries in context of object characterization.**

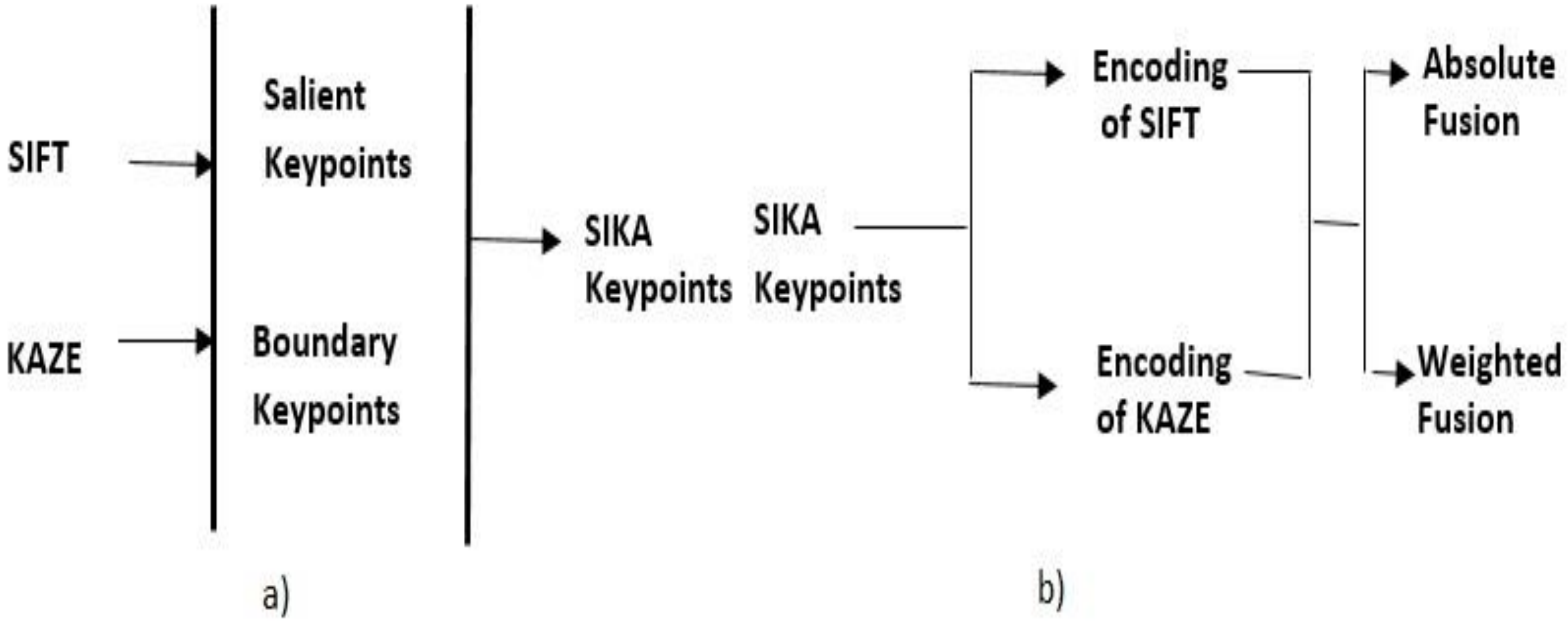
- **Adapt distinctiveness, detectability and repeatability scores [1] for keypoints to Euclidean space.**

Background

- **SIFT has been the de-facto choice for keypoint extraction.**
- **KAZE is a recent feature detection technique which exploits the non linear scale space to detect keypoints along edges and sharp discontinuities.**
- **SIKA: A combination of SIFT and KAZE keypoints has shown complementary nature of these techniques. Though it shows the effectiveness of the combination in object classification, we provide a non-heuristic approach for extracting suitable keypoints from the image with the requisite properties.**

SIKA

- **SIKA keypoints [7] are direct combination of SIFT and KAZE keypoints. The selection consists of either all or a subset of keypoints based on the available object annotations.**
- **Suited for Object Classification and similar tasks with available object annotations for training.**



SIKA

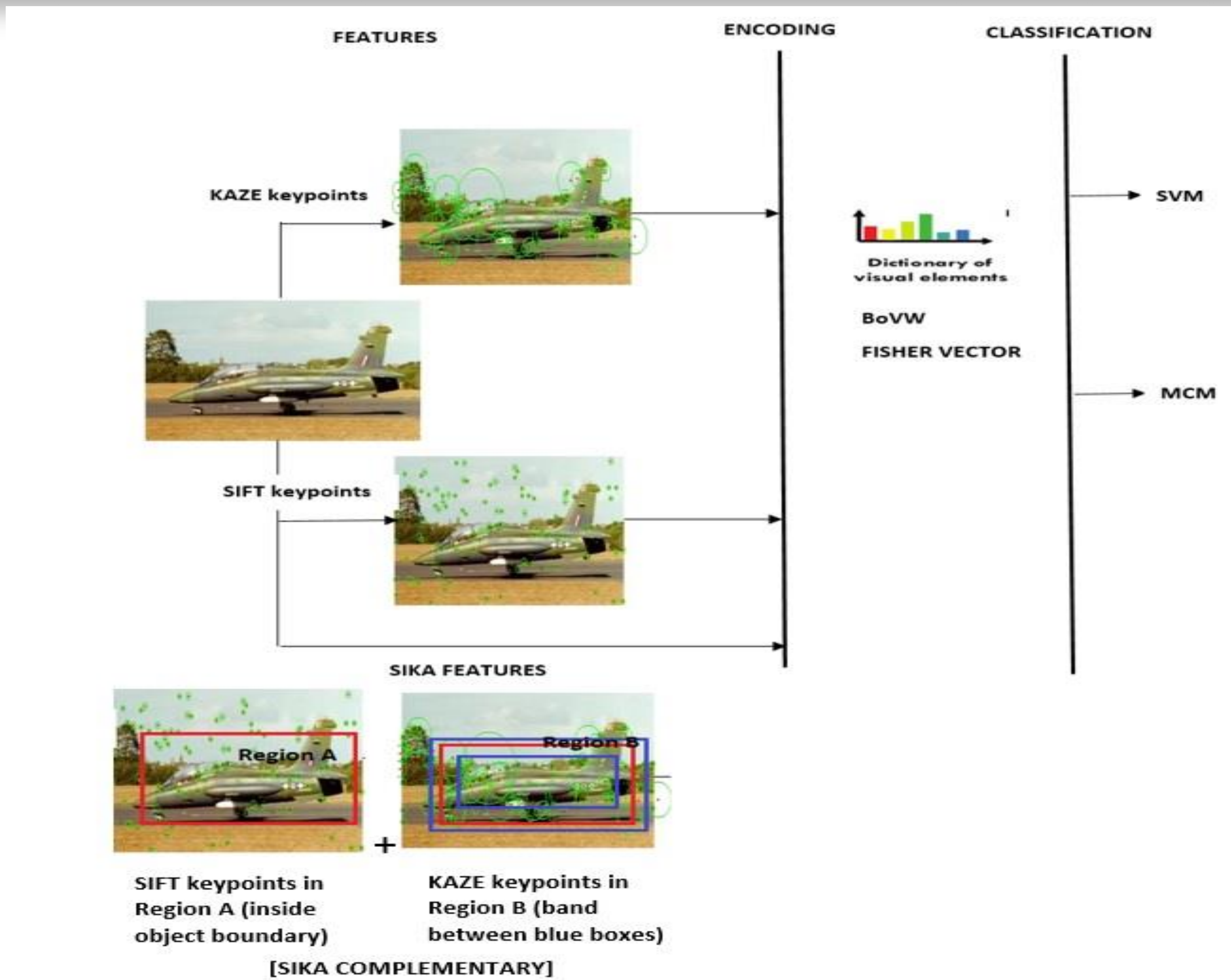
SIKA ALL

$$SIKA\ ALL_{keypoints} = SIFT_{keypoints} \cup KAZE_{keypoints}$$

SIKA Complementary

$$SIKA\ Comp_{keypoints} = SIFT_{keypoints(object)} \cup KAZE_{keypoints(boundary)}$$

SIKA: Approach



SIFT vs KAZE vs SIKA

Property	SIFT	KAZE	SIKA
Keypoint Distribution	corners	boundaries	objects
No. of Keypoints	Large	Relatively fewer	Selective (Practically needs less than 50% of keypoints as compared to SIFT and KAZE)
Scale Space	Linear	Non linear	Both
Descriptor size	128 dimensional descriptor	64/128 dimensional descriptor	Respective Descriptors
Object Classification [7]	Lags behind CNN	No where near CNN	Comparable to CNN (not always)

Proposed Methodology: An overview

1. Ranked combination: SIFT and KAZE keypoints + keypoints computed from the texture map produced by Gabor filter.
2. Sharp edges or transitions: key characteristics of objects [3]. SIFT or any other detector loses out on this crucial boundary information.

KAZE features based on non-linear anisotropic diffusion filtering [4].

3. Supplement the SIFT and KAZE keypoints from original image with the SIFT keypoints obtained from the texture map using Gabor filter. Saliency map obtained using [5] is used to threshold out 'weak' keypoints.

Proposed Methodology: Flow

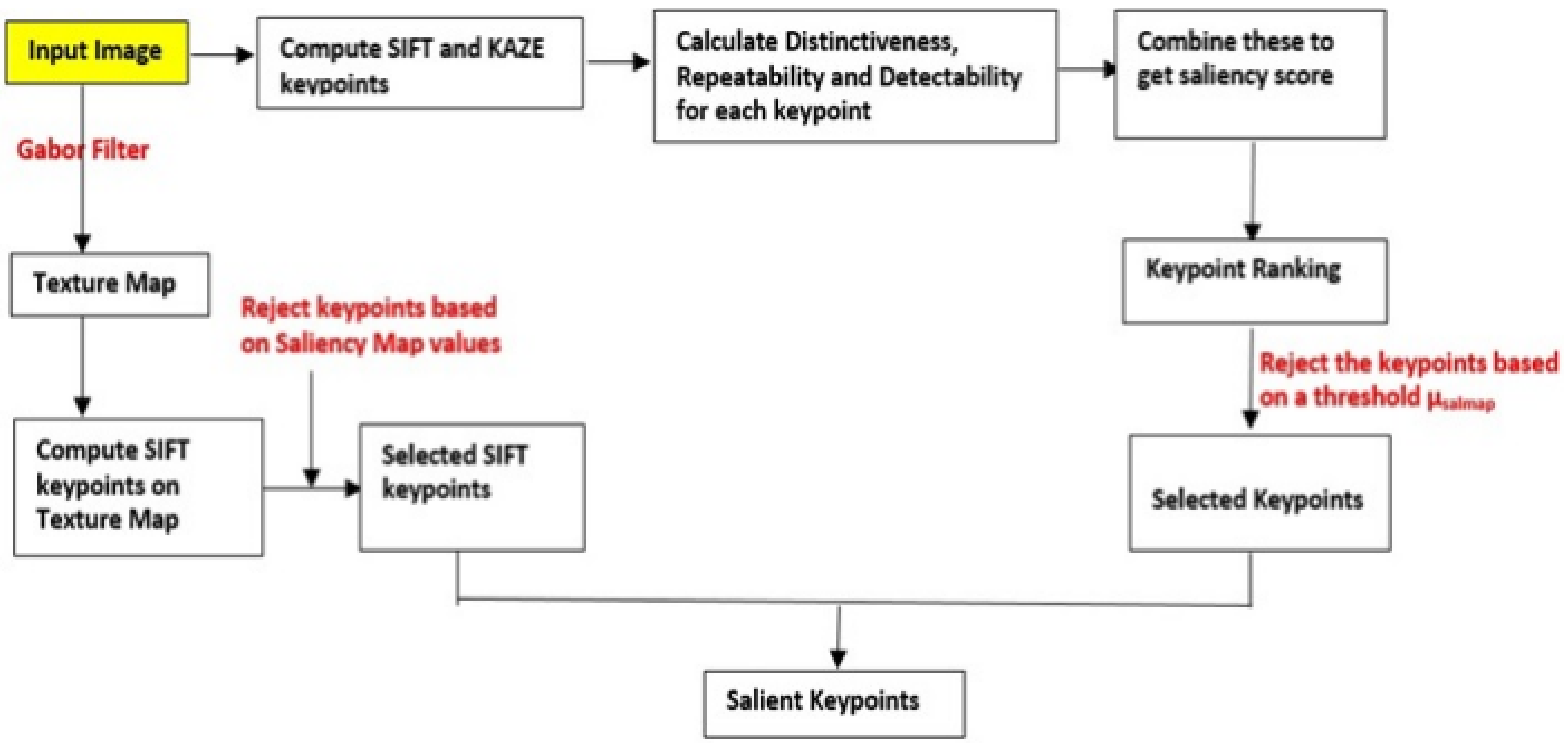


Fig 1. : Flow diagram for the proposed methodology

Keypoint Selection and Ranking

1. **Transformations:** rotation ($\pi/6$, $\pi/3$, $2 * \pi/3$), scaling (0.5, 1.5, 2), cropping (20%, 50%), affine.

$$S_{KP\,(i)} = Dist(KP(i)) + Det(KP(i)) + Rep(KP(i))$$

Where $S_{KP\,(i)}$ *: saliency score, Dist(KP(i)) : Distinctivity, Det(KP(i)) : Detectability, Rep(KP(i)) : Repeatability*

2. The description of i^{th} keypoint which gives the location (x_i , y_i) and response of the keypoint s_i .

$$KP(i) = \{(x_i\,,\, y_i),\, s_i\},\ i = 1 \ldots N$$

Keypoint Selection and Ranking

3. Distinctiveness gives the summation of the Euclidean distances between every pair of keypoint descriptors in the same image.

$$Dist(KP(i)) = \frac{1}{N-1} \sum_{(x_i, y_i) \epsilon KP(i), i \neq j} ED(d_i, d_j)$$

Keypoint Selection and Ranking

4. Repeatability gives Euclidean distance (ED) between the keypoint descriptor in the original image to the keypoint descriptor mapped in the corresponding transform, t. Here, nTransf is the number of transformations.

$$Rep(KP(i)) = \frac{1}{nTransf} \sum_{t=1}^{nTransf} ED(d_i, d_i^t)$$

Keypoint Selection and Ranking

5. Detectability gives the summation of the strengths of the keypoint in the original image and its respective transforms.

$$Det(KP(i)) = \frac{1}{nTransf} \sum_{t=1}^{nTransf} s_{it}$$

Keypoint Selection and Ranking

6. We select the KAZE and SIFT keypoints which have saliency score greater than the respective mean saliency scores.

$$SalientKP = KP(i) \quad s.t. \quad S_{KP(i)} \geq \mu_{salscore}, 1 \leq i \leq N$$

where N is the total count of keypoint from respective detector and $\mu_{salscore}$ is mean of the saliency scores.

$$\mu_{salscore} = mean(S_{KP(i)}), \qquad 1 \leq i \leq N$$

Texture Map based SIFT keypoints

1. **SIFT keypoints are calculated on the original image. Then, the orientation histogram of the keypoints is constructed. The dominant orientations are found by binning the keypoint orientations into prespecified number of bins. The image is then convolved with Gabor filter using these dominant orientations.**

$$G_{\theta,u,\sigma}(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} . e^{2\pi\imath(ux\cos\theta + uy\sin\theta)}$$

where u denotes the frequency of the sinusoidal function, θ gives the orientation of the function, σ is the standard deviation of the Gaussian function.

Texture Map based SIFT keypoints

2. Next, the saliency map [5] is calculated for the original image. For each keypoint, if the saliency value is greater than the mean saliency then the keypoint is retained.

$$TextureKP = KP(i) \quad s.t. \quad S_{KP(i)} \geq \mu_{salmap}, 1 \leq i \leq N$$

where TextureKP denotes the set of keypoints which are salient for representing the texture. μ_{salmap} denotes the mean of the saliency map.

Algorithm: Ranking Salient keypoints

Algorithm 1 Algorithm for Ranking Salient Keypoints

procedure RANK–KP
    Compute SIFT ($SIFT_{KP}$) and KAZE ($KAZE_{KP}$) keypoints on input image I
    for each keypoint i
        $S_{KP(i)} = Dist(KP(i)) + Det(KP(i)) + Rep(KP(i))$
    end for
    $SalientKP = \{[SIFT_{KP} \quad KAZE_{KP}] \;\mid\; S_{KP(i)} \geq \mu_{salscore}\})$
    Distribute orientations of $SIFT_{KP}$ into c equal sized bins
    Compute texture map using Gabor Filter with c orientations
    $TextureKP = \{SIFT_{KP} \;\mid\; S_{KP(i)} \geq \mu_{salmap}\}$
    $SalientKP = [SalientKP \quad TextureKP]$
end procedure

EXPERIMENTAL RESULTS AND DISCUSSIONS

Datasets:

- **Caltech 101: to show the effectiveness of the algo. that the salient keypoints characterize and represent the objects.**
- **VGG affine dataset: for object matching.**

Object Representation

TABLE I: Performance Analysis of various feature detectors for object representation.

Feature Detector	Keypoints inside Bounding Box (in %) (a)	Keypoints inside region (in %) (b)
SIFT	76	62
SURF	71	58
KAZE	87.62	69
RankedKP	84	82.7

Object Representation

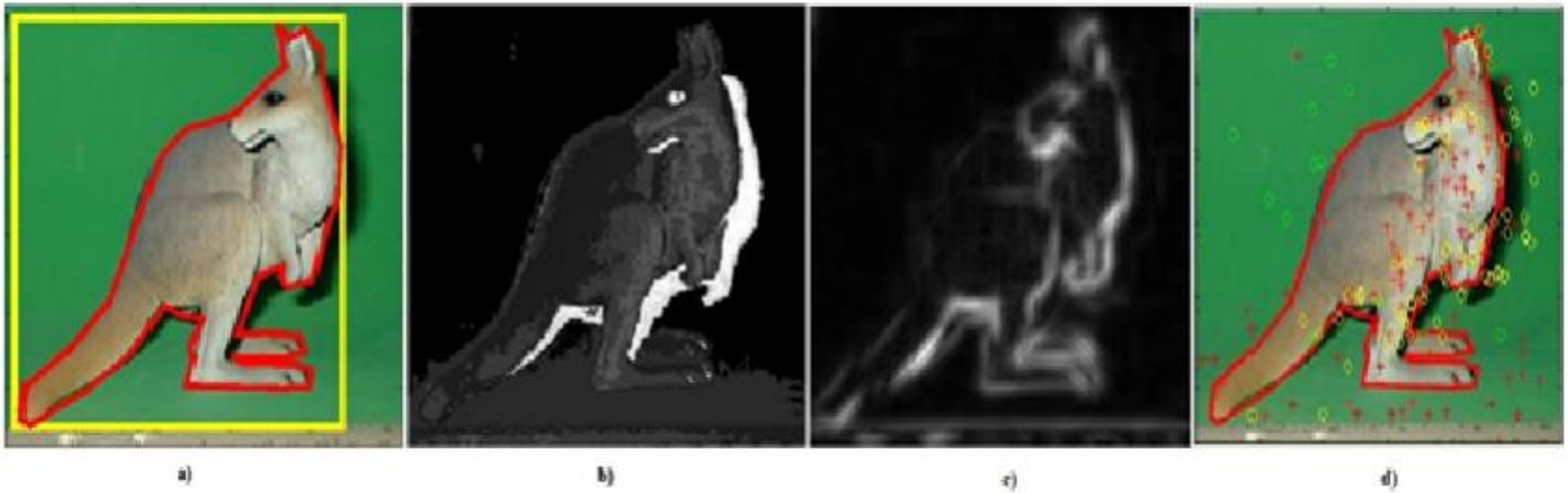


Fig. 2: Figure showing a) Object annotation b) Saliency Map c) Gabor filtered image (Texture Map) d) Ranked keypoints inside the object contour

Object Representation

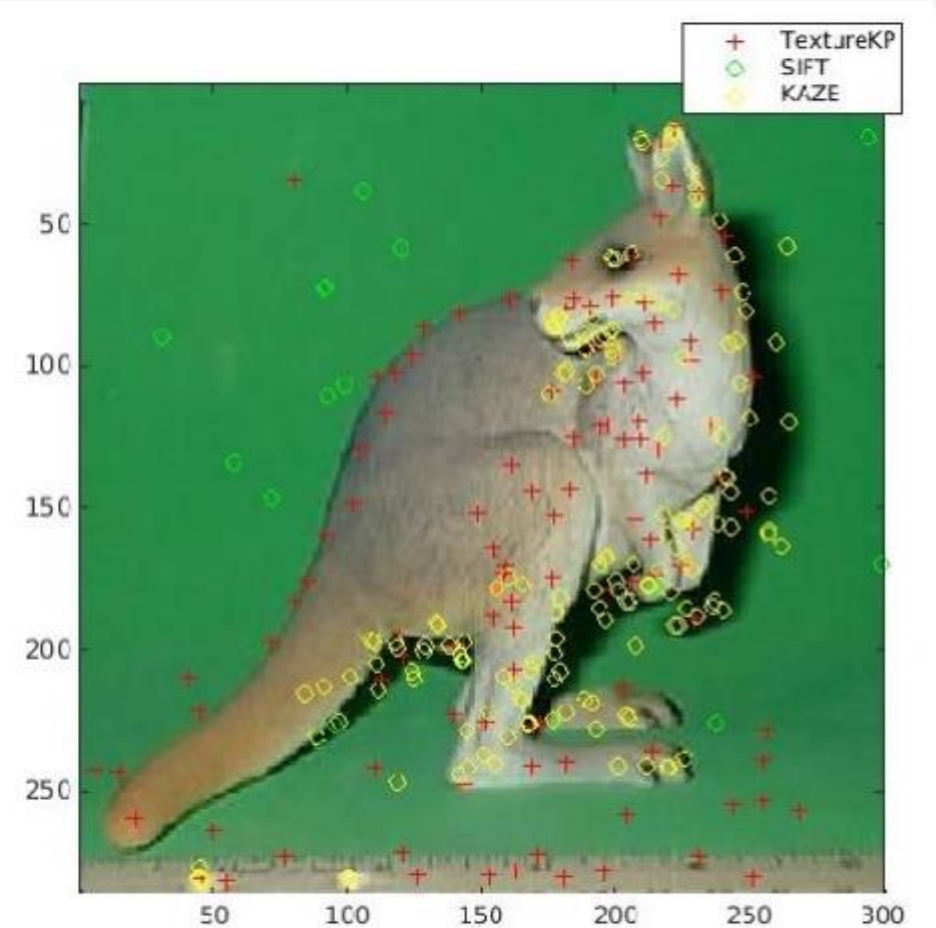


Fig. 3: Texture and Ranked (SIFT and KAZE) keypoints

Object Matching

TABLE II: Object Matching.

Feature Detector	Total KP matches (%) (a)	Correct KP Matches(%) (b)	Mean ED (c)
SIFT	67	90	0.0960
SURF	78	74	0.0752
KAZE	81	91	0.1297
(Ranked+Texture)	89	96	0.0011

Object Matching

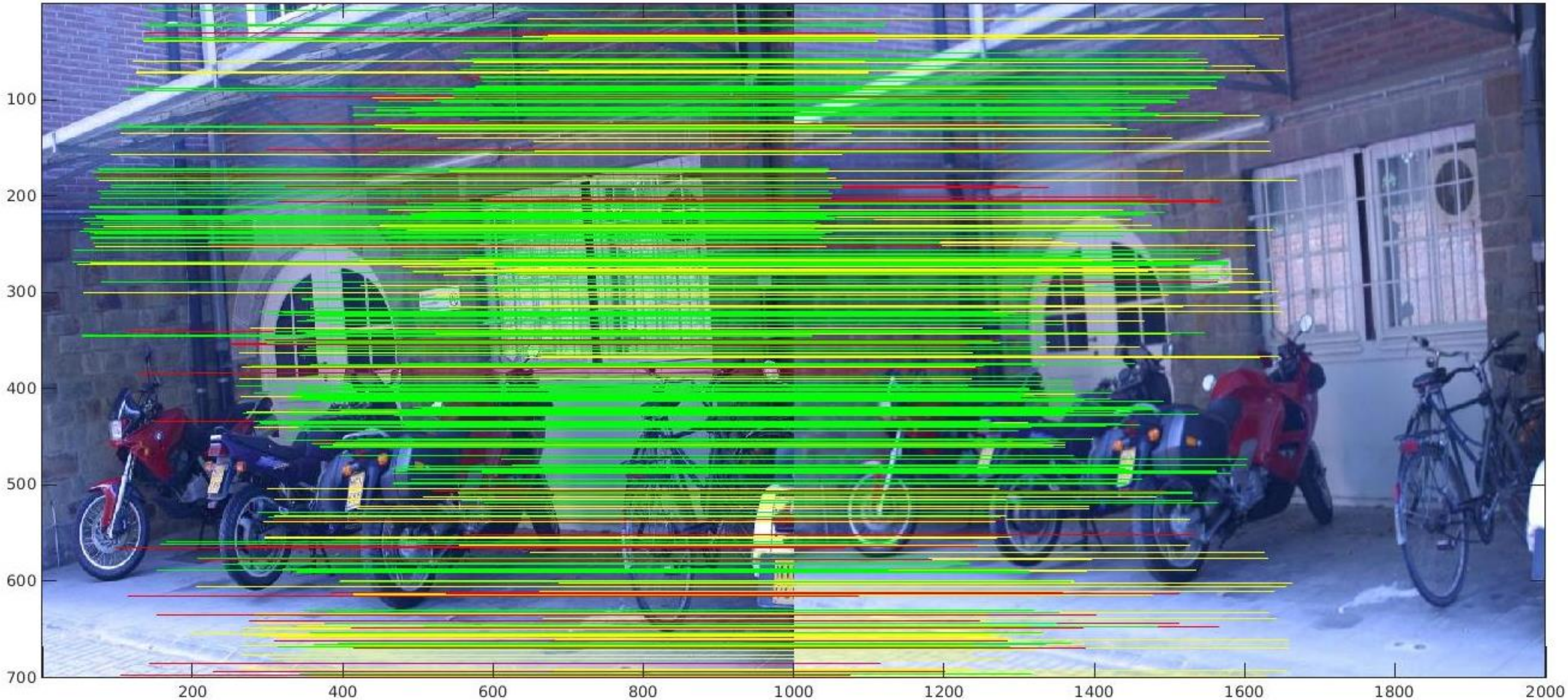


Fig. 4: Correctly matched keypoints by the proposed selection strategy: red (KAZE), yellow (SIFT), green (TextureKP) on the bikes dataset (VGG).

Object Matching

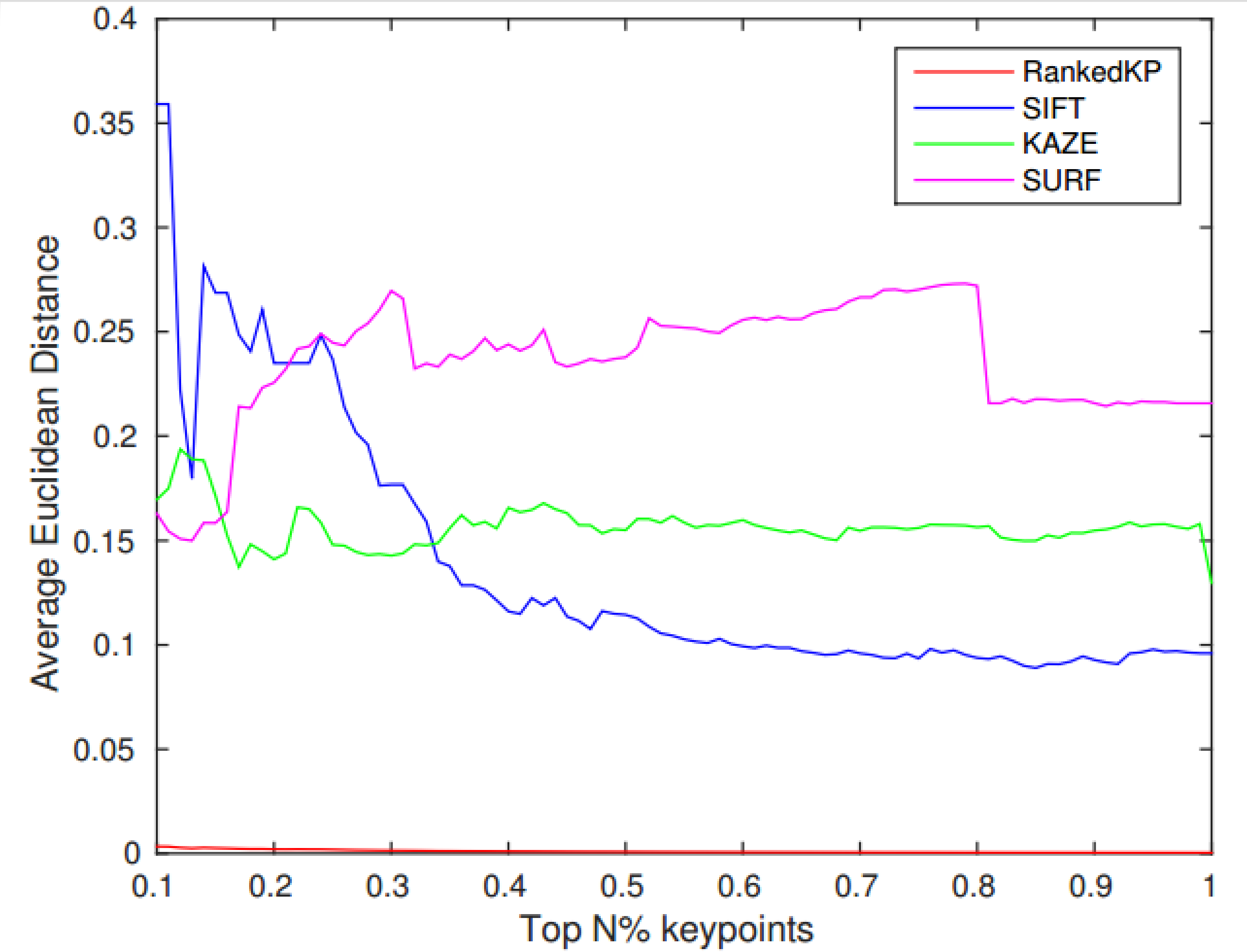


Fig. 5: Average ED vs top N% keypoints of the feature set

CONCLUSION

- **Novel keypoint selection scheme based on SIFT and KAZE proposed. The technique incorporated texture information by finding SIFT keypoints on a texture map (using Gabor).**
- **Technique can characterize an object region more efficiently than other contemporary detectors.**
- **Less prone to false positives.**
- **It will help in extending the existing object matching and classification algorithms.**
- **Practical applications: object localization, segmentation and many other domains.**
- **Holds promise to extend the existing state of the art in many application areas where objects are involved**

Bibliography

[1] W. Hartmann, M. Havlena, and K. Schindler, “Predicting matchability,” in Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on. IEEE, 2014, pp. 9–16.

[2] S. Buoncompagni, D. Maio, D. Maltoni, and S. Papi, “Saliency-based keypoint selection for fast object detection and matching,” Pattern Recognition Letters, 2015.

[3] B. Alexe, T. Deselaers, and V. Ferrari, “What is an object?” in Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on. IEEE, 2010, pp. 73–80.

[4] P. Perona and J. Malik, “Scale-space and edge detection using anisotropic diffusion,” Pattern Analysis and Machine Intelligence, IEEE Transactions on, vol. 12, no. 7, pp. 629–639, 1990.

[5] P. Mukherjee, B. Lall, and A. Shah, “Saliency map based improved segmentation,” in Image Processing (ICIP), 2015 IEEE International Conference on (Accepted). IEEE, 2015.

[6] P. Alcantarilla, A. Bartoli and A. Davison, “Kaze Features,” *In Proceedings of the 12th European conference on Computer Vision*, vol. 6, pp. 214-227, 2012.

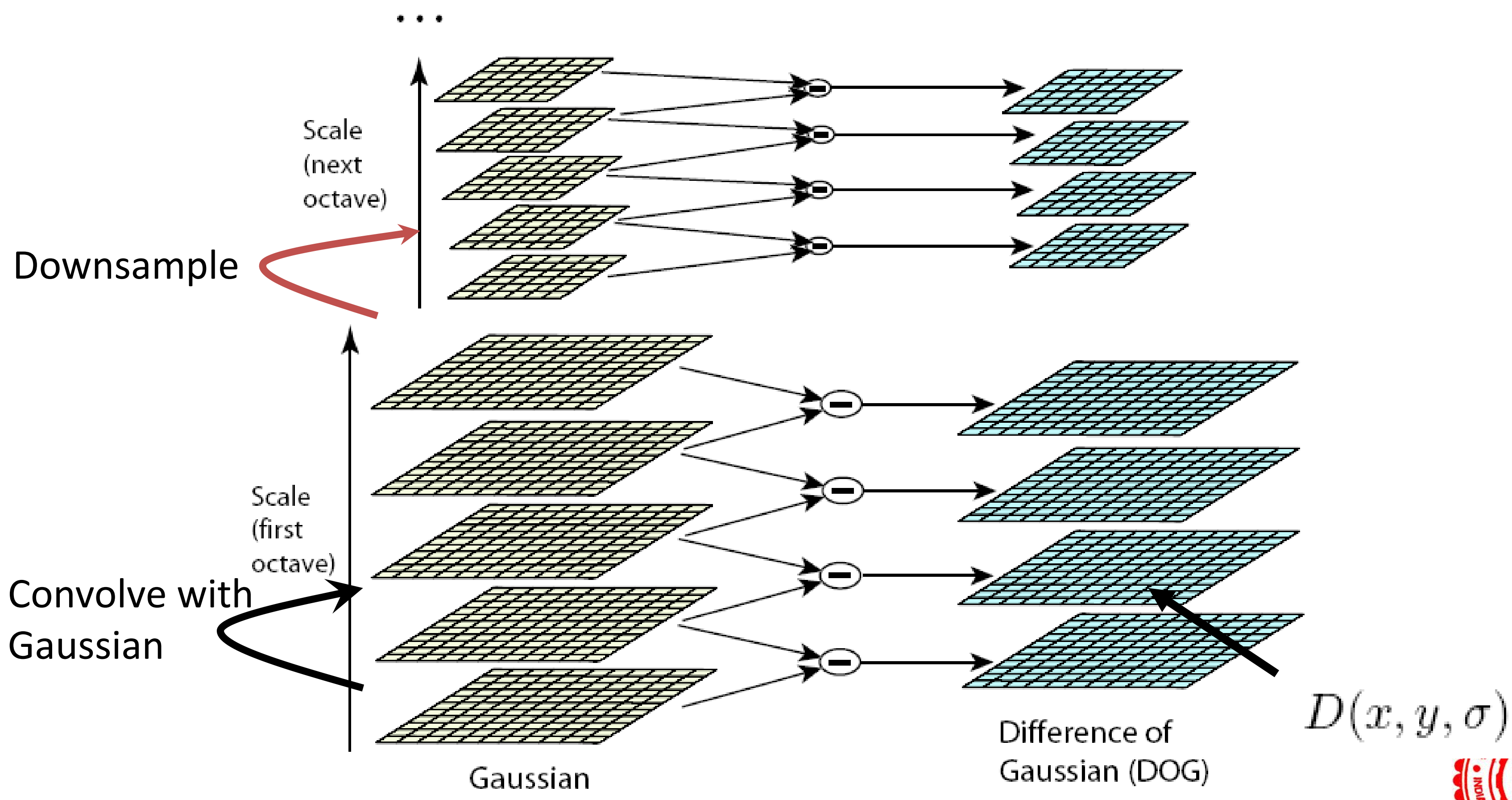
[7] Srivastava, Siddharth, Prerana Mukherjee, and Brejesh Lall. "Characterizing objects with SIKA features for multiclass classification." *Applied Soft Computing* (2015).

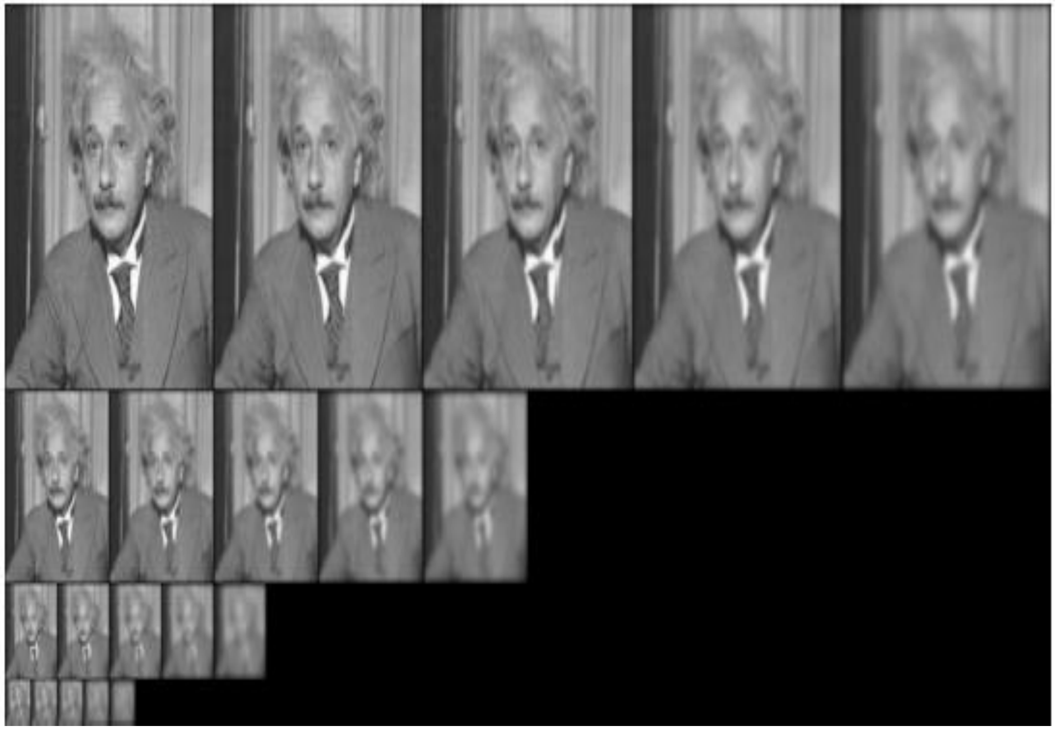
Thank-you!!!

Appendix

Scale Invariant Feature Transform: Keypoint Detection

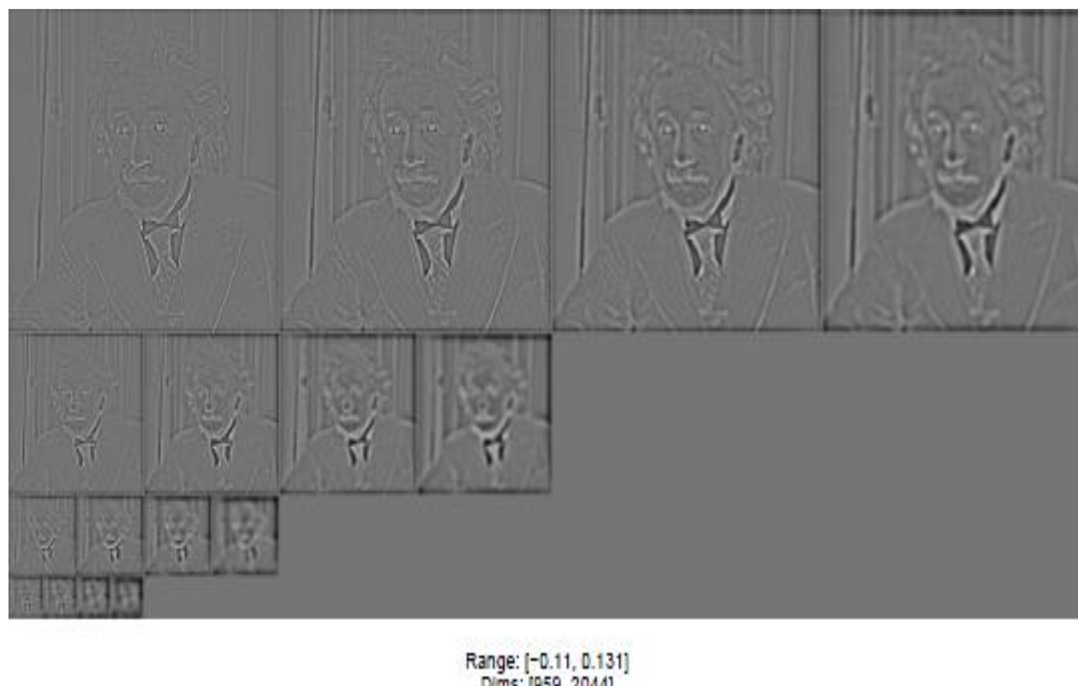
Step 1: Construction of Scale Space





Gaussian images grouped by octave.

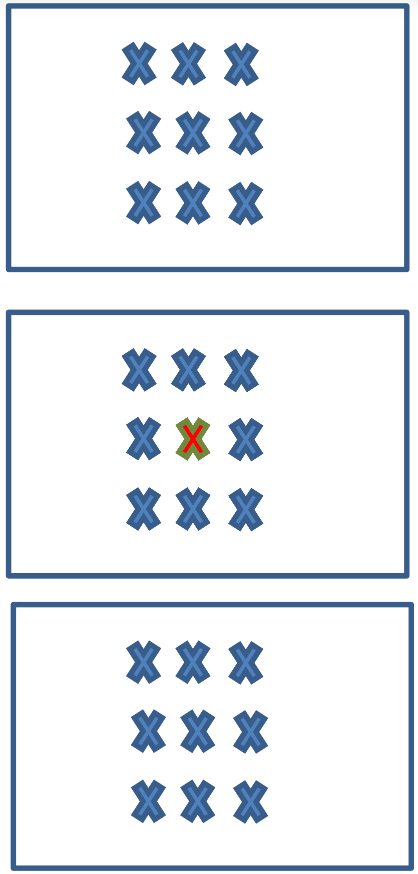
DoG images grouped by octave



Extrema Detection (for each pixel)

Choose consecutive DoG images

26 neighbours



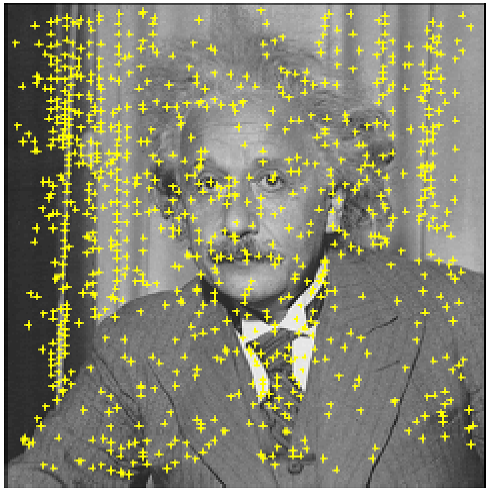
Optimization Tricks:

1. For non-maxima and non-minima all points need not to be compared
2. First and last images in the octave need not be compared

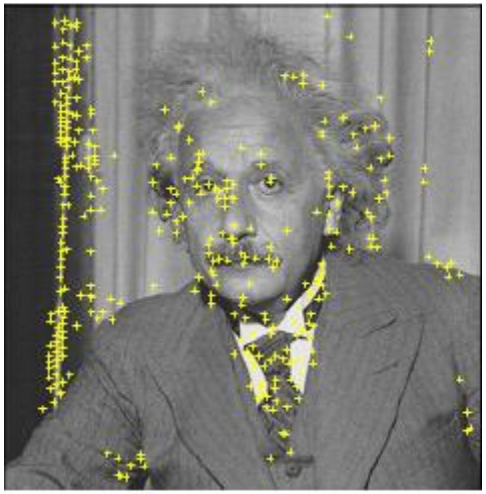
Take pixel if it is local maxima/local minima than all of them.
This is called a **KEYPOINT**.

Step II: Keypoint Localization

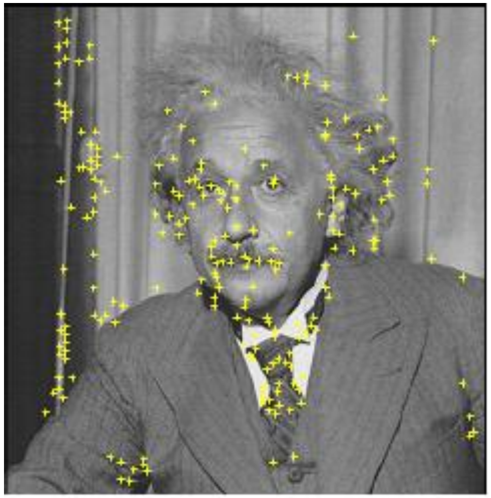
- (b) Reject keypoints with low contrast
- (c) Reject keypoints that are localized along an edge



a)



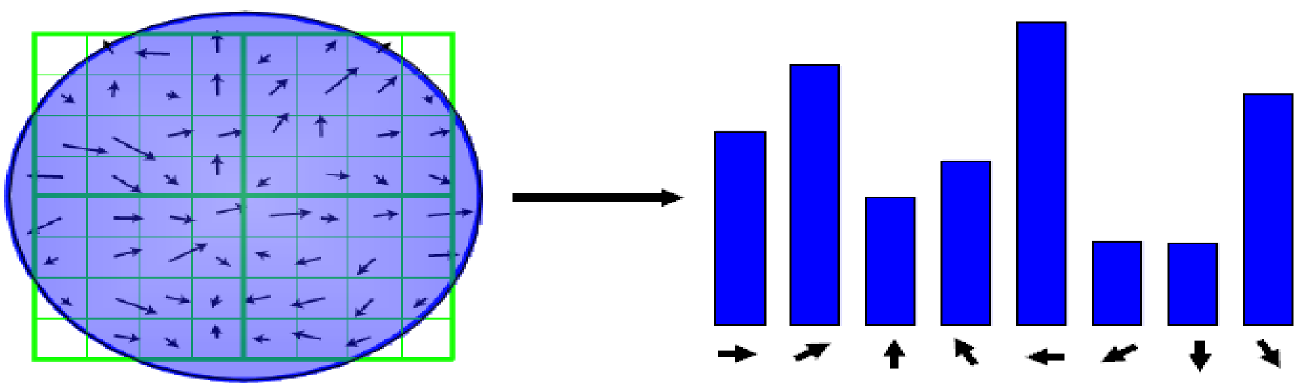
b)



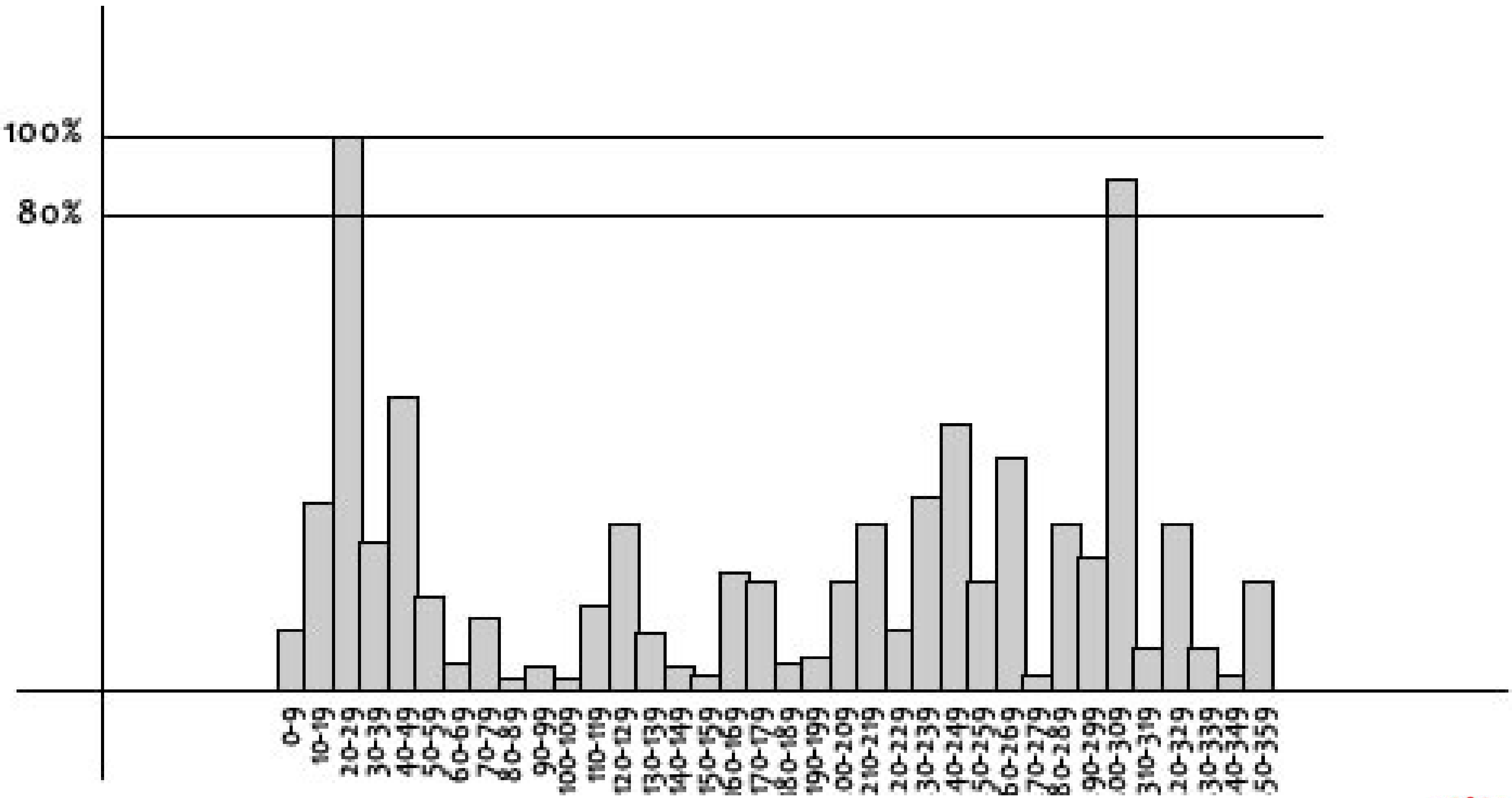
c)

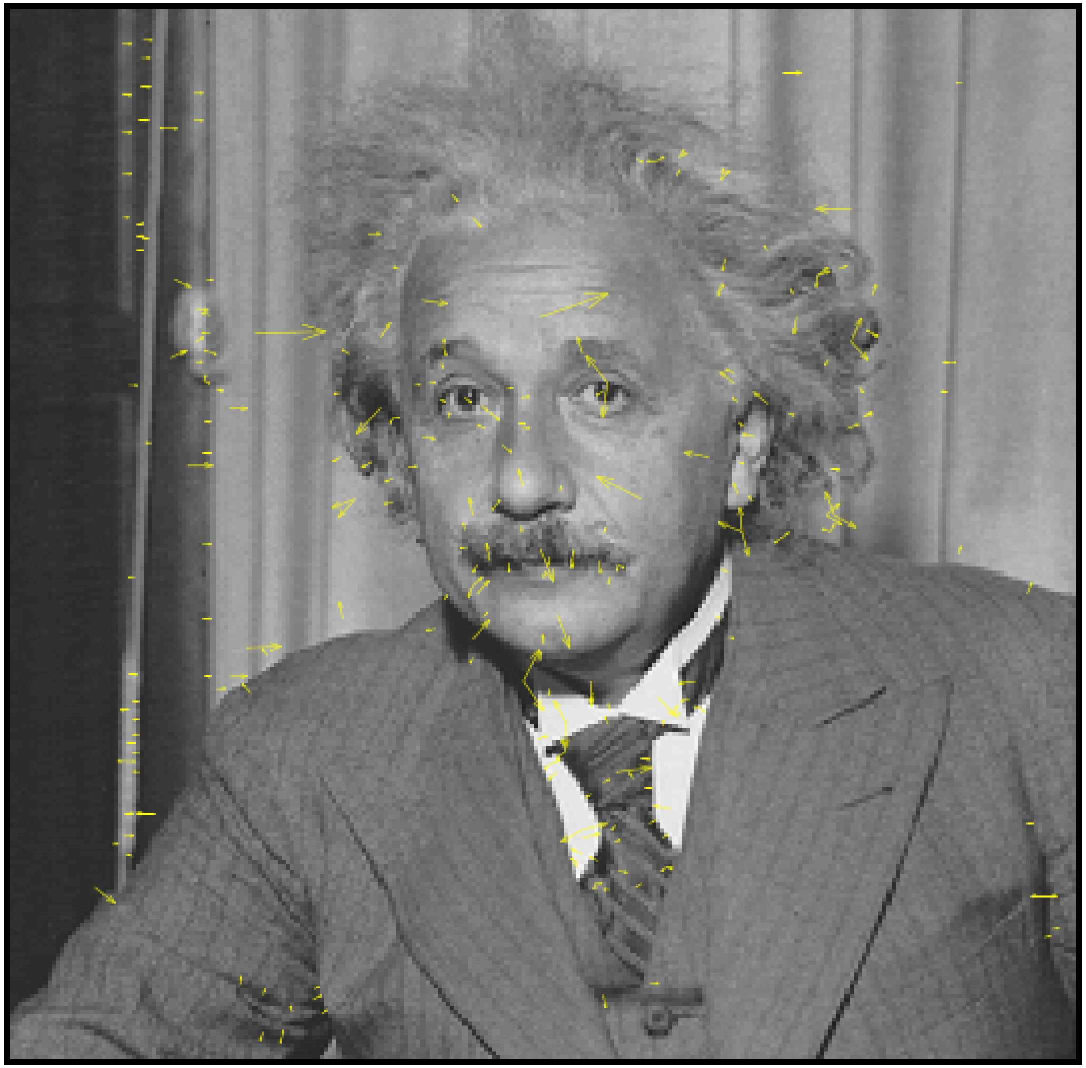
Step III: Orientation Assignment

- Create gradient histogram for the keypoint neighbourhood (36 bins)
- Neighborhood: a circular Gaussian falloff from the keypoint center (\sigma=1.5 pixels at the current scale, so the effective neighborhood is about 9x9)



Orientation Assignment (Contd…)

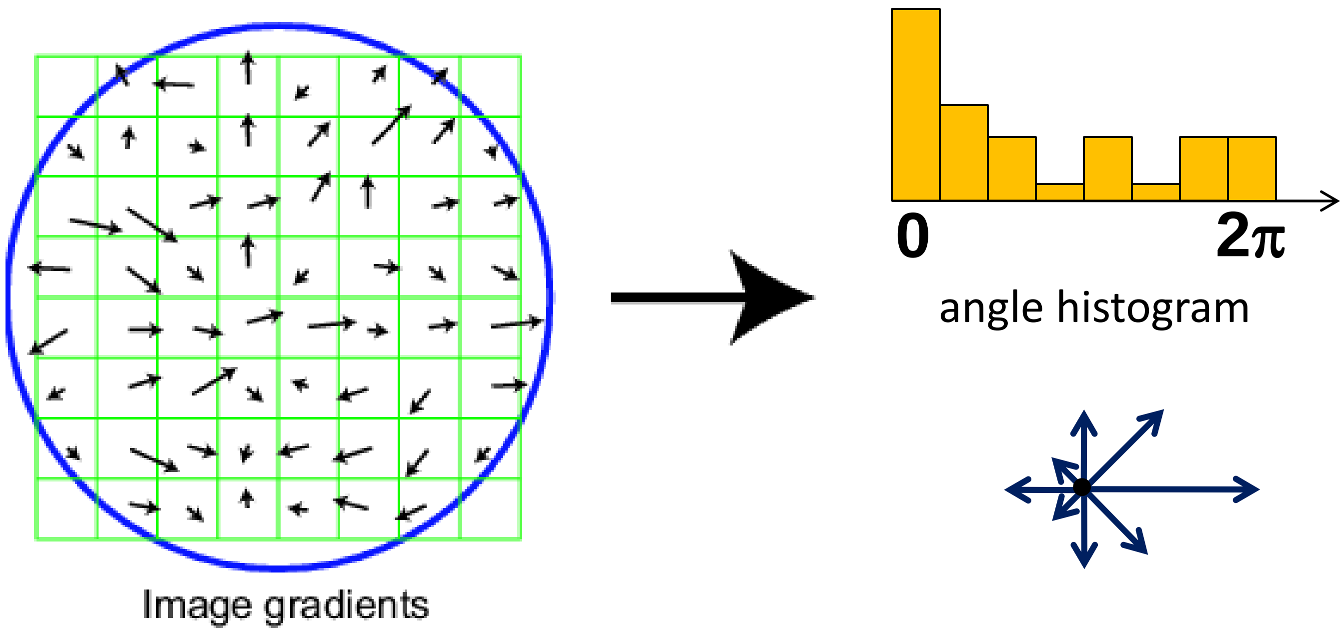




Extracted keypoints, arrows indicating scale and orientation

Scale Invariant Feature Transform: Keypoint Description

- Take 16x16 square window around detected keypoint
- Decompose this into 4x4 tiles
- Compute gradient orientation for each pixel (8 bins)
- Create histogram over edge orientations weighted by magnitude



4x4x8= 128D

Adapted from slide by David Lowe

KAZE: Background

Nonlinear Diffusion Filtering

- **Nonlinear diffusion approaches** describe the **evolution of the luminance of an image** through **increasing scale levels** as the divergence of a **flow** function that controls the diffusion process

$$\frac{\partial L}{\partial t} = div\left(c\left(x, y, t\right) \cdot \nabla L\right) \qquad ($$

- The function c depends on the **local image differential structure**, and this function can be either a **scalar** or a **tensor**
- The time t is the **scale parameter**, and larger values lead to simpler image representations

KAZE: Background

Perona and Malik Diffusion Equation

- Function c dependent on the **gradient magnitude**
- **Reduce diffusion at edges location**, encouraging smoothing within a region instead of smoothing across boundaries

$$c(x, y, t) = g(|\nabla L_\sigma(x, y, t)|)$$

- Two different formulations for the **conductivity function** g
  - g_1 promotes **high-contrast edges**
  - g_2 promotes **wide regions over smaller ones**

$$g_1 = \exp\left(-\frac{|\nabla L_\sigma|^2}{k^2}\right)$$

$$g_2 = \frac{1}{1+\frac{|\nabla L_\sigma|^2}{k^2}}$$

KAZE: Background

- The **contrast factor** k is computed **empirically** as the 70% percentile of the gradient histogram of a smoothed version of the original images
  - It can be also set by hand or by some learning
- If the **conductivity function** c is **constant**, we obtain the **heat equation**, i.e. **linear diffusion**
- **Rapidly decreasing diffusivities**
  - Smoothing on both sides of an edge is much stronger than smoothing across it

$$g_3 = \begin{cases} 1 & , \quad |\nabla L_\sigma|^2 = 0 \\ 1 - \exp\left(-\frac{3.315}{(|\nabla L_\sigma|/k)^8}\right) & , \quad |\nabla L_\sigma|^2 > 0 \end{cases} \tag{9}$$

KAZE: Background

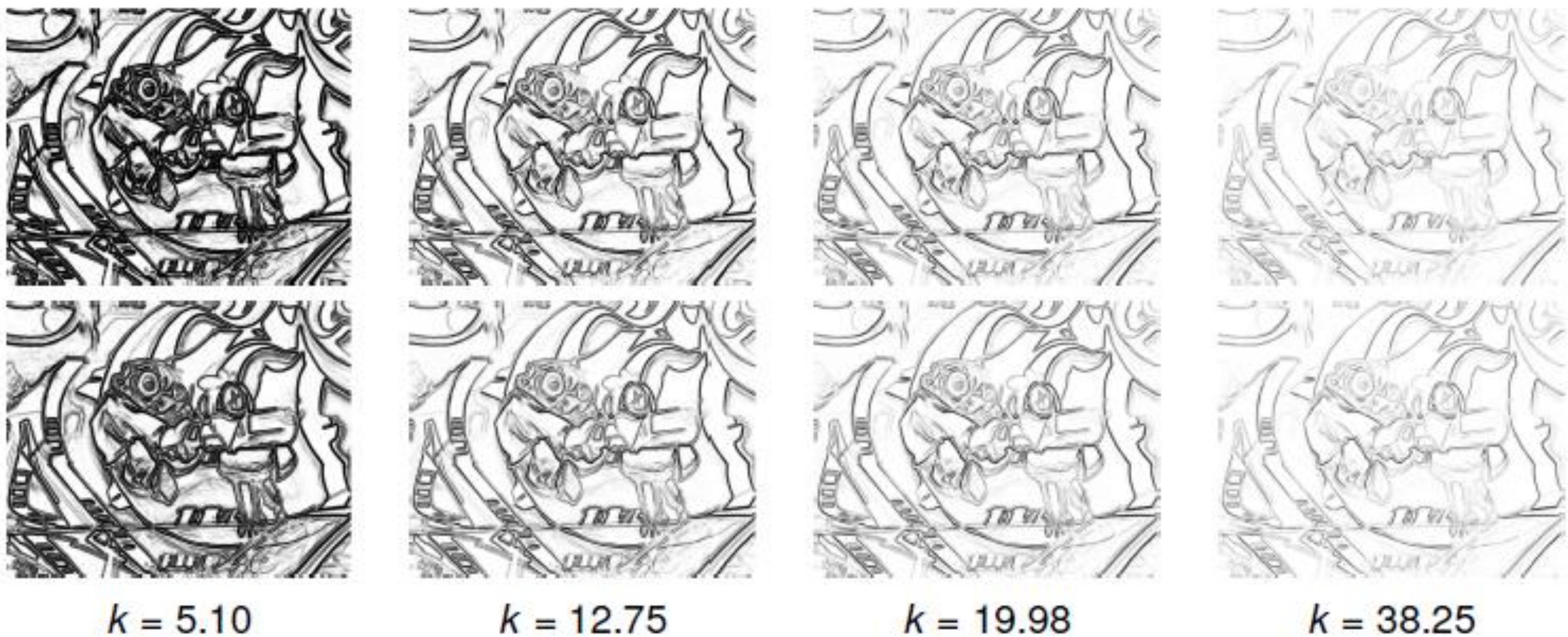


Figure: First row: g_1 conductivity function. Second row: g_2 conductivity function. Notice that for increasing values of k only higher gradients are considered.

KAZE: Background

Additive Operator Splitting (AOS)

- Modification of the **semi-implicit scheme**

$$\frac{L^{i+1} - L^i}{\tau} = \sum_{l=1}^{m} A_l(L^i)L^{i+1} \ ,$$

- The solution L^{i+1} can be obtained as:

$$L^{i+1} = \left(I - \tau \sum_{l=1}^{m} A_l(L^i) \right)^{-1} L^i$$

- Now the total diffusion is the **addition of two 1D diffusion processes**
- The matrix A_l encodes the diffusivities for each image dimension

KAZE: Keypoint Detection

- Steps in **KAZE Detection**:
  - Build **nonlinear scale space** using **AOS** and a set of **octaves** O and **sublevels** S

$$\sigma_i(o,s) = \sigma_0 2^{o+s/S},\ o \in [0 \dots O-1],\ s \in [0 \dots S-1],\ i \in [0 \dots N]$$

  - We need to map **scale units** to **time units**:

$$t_i = \frac{1}{2}\sigma_i^2,\ i = \{0 \dots N\}$$

$$L^{i+1} = \left(I - (t_{i+1} - t_i) \cdot \sum_{l=1}^{m} A_l\left(L^i\right)\right)^{-1} L^i\ .$$

equation for building non linear scale space using AOS

Non linear vs linear scale space

Comparison between gaussian blurring and nonlinear diffusion

$t_i = 5.12$ $t_i = 20.48$ $t_i = 81.92$ $t_i = 130.04$ $t_i = 206.42$

KAZE: Keypoint Detection

$$L_{Hessian} = \sigma^2 \left(L_{xx} L_{yy} - L_{xy}^2 \right)$$

we analyze the detector response at different scale levels σ_i. We search for maxima in scale and spatial location. The search for extrema is performed in all the filtered images except $i = 0$ and $i = N$. Each extrema is searched over a rectangular window of size $\sigma_i \times \sigma_i$ on the current i, upper $i + 1$ and lower $i - 1$ filtered images.

The set of first and second order derivatives are approximated by means of 3×3 Scharr filters of different derivative step sizes σ_i. Second order derivatives are approximated by using consecutive Scharr filters in the desired coordinates of the derivatives.

KAZE: Keypoint Detection

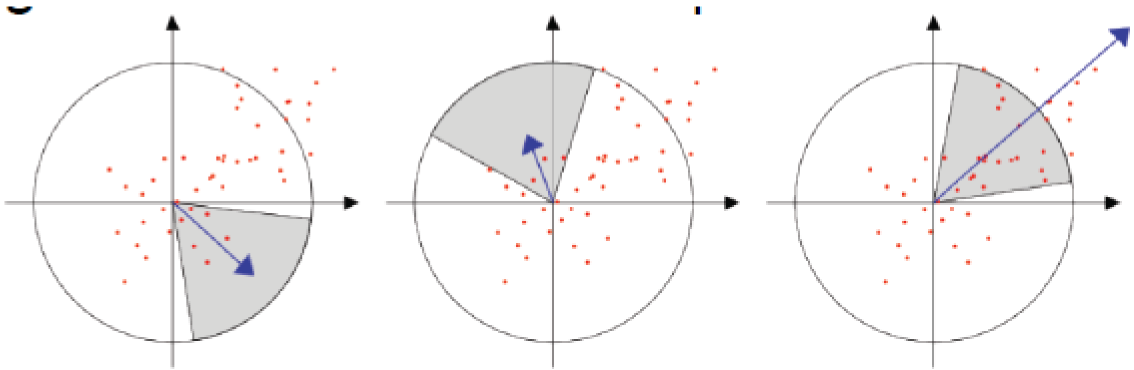
Scharr edge filter

The Scharr operator is the most common technique with two kernels used to estimate the two dimensional second derivatives horizontally and vertically. The operator for the two direction is given by the following formula:

$$K_x = \begin{pmatrix} 3 & 0 & -3 \\ 10 & 0 & -10 \\ 3 & 0 & -3 \end{pmatrix}, K_y = \begin{pmatrix} 3 & 10 & 3 \\ 0 & 0 & 0 \\ -3 & -10 & -3 \end{pmatrix}.$$

KAZE: Keypoint Description

Similar to SURF, we find the dominant orientation in a circular area of radius $6\sigma_i$ with a sampling step of size σ_i. For each of the samples in the circular area, first order derivatives L_x and L_y are weighted with a Gaussian centered at the interest point. Then, the derivative responses are represented as points in vector space and the dominant orientation is found by summing the responses within a sliding circle segment covering an angle of $\pi/3$. From the longest vector the dominant orientation is obtained.



KAZE: Keypoint Description

Building the Descriptor. We use the M-SURF descriptor adapted to our nonlinear scale space framework. For a detected feature at scale σ_i, first order derivatives L_x and L_y of size σ_i are computed over a $24\sigma_i \times 24\sigma_i$ rectangular grid. This grid is divided into 4×4 subregions of size $9\sigma_i \times 9\sigma_i$ with an overlap of $2\sigma_i$. The derivative responses in each subregion are weighted with a Gaussian ($\sigma_1 = 2.5\sigma_i$) centered on the subregion center and summed into a descriptor vector $d_v = (\sum L_x, \sum L_y, \sum |L_x|, \sum |L_y|)$. Then, each subregion vector is weighted using a Gaussian ($\sigma_2 = 1.5\sigma_i$) defined over a mask of 4×4 and centered on the interest keypoint. When considering the dominant orientation of the keypoint, each of the samples in the rectangular grid is rotated according to the dominant orientation. In addition, the derivatives are also computed according to the dominant orientation. Finally, the descriptor vector of length 64 is normalized into a unit vector to achieve invariance to contrast.