Corner and Interest Point Detection

Lecture Notes

September 26, 2011

Much of computer vision in the past decade has come to rely on isolated features which are invariant under image translation, rotation, moderate noise, moderate scaling, and somewhat to changes in view points. These features have come to be known as **key points**, **interest points**, and others. These features, in contrast to edges and contours which are localized in one direction and vary along the other, are localized in both dimensions and are generically isolated from each other. The key points can be computed either as a property of the image geometry/shape content or from its appearance. For example, corners (L-Junctions), T-Junctions, Y-Junctions, X-Junctions, as well as extrema and reflection points of curves [Asada and Brady 1986, Mokhtarian and Macworth 1986], or of ridges [Shilat etal CVPR97] are isolated features computed from the geometric content of the image. On the other hand, SIFT points [Lowe: IJCV04] are points satisfying a differential conditions on the local appearance of the image. These lecture notes are a partial coverage of the history that often gets ignored in the coverage of the background.

1 Corner Detection: Moravec and Harris - Stephens Algorithm

One of the earliest appearance-based interest point detections [Moravec:IJCAI:1977] detects points exceeding some threshold of the auto correlation signal in all four directions. Maxima of the determinant of the Hessian $|H| = f_{xx}f_{yy} - f_{xy}$ are used by [Beandet:IJCPR:1978]. Maxima of normalized iso-intensity level set curvature are used by [Kitchen:Rosenfeld:1982]. [Nagel 1983] shows the above two measures are identical.

Quoting Moravec [Moravec 1976] "The interest operator locates small patches, called features, scattered more or less uniformly over images and having the property that the corresponding points are likely unambiguously findable in subsequent images". The goal of this interest operator is to find local maxima of a directional variance measure which avoids featureless areas and simple edges with no variance along the edge. This directional variance is the minimum over all directions of sums of squares of differences of each pixel in a window

from the one adjacent to it in each given direction. A window is a local maximum if it is larger than its neighbors.

Specifically, let $f(x, y)$ denote the image, and let W denote the window over which measurements are averaged. Then define,

$$J(x, y, \xi, \eta) = \sum_{\Delta y=-W_y}^{W_y} \sum_{\Delta x=-W_x}^{W_x} \left[f(x + \Delta x + \xi, y + \Delta y + \eta) - f(x + \Delta x, y + \Delta y) \right]^2, \qquad (1)$$

where $(\xi, \eta) \in \{(1, 0), (1, 1), (0, 1), (-1, 1)\}$ represents one of the four image directions: East, North-East, North, North-West, and W_x and W_y define the half-width and half-height of the window. We can rewrite this using a convolution notation by defining a kernel,

$$h(\Delta x, \Delta y) = \begin{cases} 1 & \text{if } |\Delta x| \leq W_x \ \ \& \ \ |\Delta y| \leq W_y \\ 0 & \text{otherwise,} \end{cases}$$

which is a rectangular pulse defined over the window W.

Also define a function to represent the square of the directional derivative in the direction (ξ, η),

$$F(x, y; \xi, \eta) = \left[f(x + \xi, y + \eta) - f(x, y) \right]^2.$$

Then,

$$\begin{aligned} J(x, y; \xi, \eta) &= \sum_{\Delta y=-\infty}^{\infty} \sum_{\Delta x=-\infty}^{\infty} h(\Delta x, \Delta y) F(x - \Delta x, y - \Delta y; \xi, \eta) \\ &= h(x, y) * F(x, y; \xi, \eta), \end{aligned}$$

which gives the interpretation that J is a smoothing of F by a rectangular pulse kernel h, namely, the square of the directional derivative of f in the direction (ξ, η). That a Gaussian kernel does a better job of smoothing was one of the improvements suggested by Harris and Stephens [Harris:Stephens:1988]. The first order approximation of $f(x, y)$ also gives a first order approximation of F. Writing the directional derivative of f as a sum of f_x and f_y, we can write,

$$\begin{aligned} F(x, y, \xi, \eta) &= \left[f(x, y) + f_x \xi + f_y \eta - f(x, y) \right]^2 + O(\xi^3, \eta^3) \\ &= f_x^2 \xi^2 + 2 f_x f_y \xi \eta + f_y^2 \eta^2 + O(\xi^3, \eta^3) \\ &= \begin{bmatrix} \xi & \eta \end{bmatrix} \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix} + O(\xi^3, \eta^3). \end{aligned}$$

So that,

$$\begin{aligned} J(x,y;\xi,\eta) &= h(x,y) * F(x,y;\xi,\eta) \\ &= \begin{bmatrix} \xi & \eta \end{bmatrix} \begin{bmatrix} h*f_x^2 & h*f_xf_y \\ h*f_xf_y & h*f_y^2 \end{bmatrix} \begin{bmatrix} \xi \\ \eta \end{bmatrix} + O(\xi^3,\eta^3) \\ &= \begin{bmatrix} \xi & \eta \end{bmatrix} M \begin{bmatrix} \xi \\ \eta \end{bmatrix} + O(\xi^3,\eta^3), \end{aligned}$$

where

$$M = \begin{bmatrix} h*f_x^2 & h*f_xf_y \\ h*f_xf_y & h*f_y^2 \end{bmatrix}.$$

The original Moravec's algorithm then finds

$$\overline{J}(x,y) = \min_{(\xi,\eta)\in\{(1,0),(1,1),(0,1),(-1,1)\}} J(x,y;\xi,\eta),$$

and then finds the local maxima of $\overline{J}$. The recognition that J can be written as a quadric form defined by a matrix of first-order moments implies that not only the original four discrete directions of Moravec but all directions can be explored through the analytic expansion, thus reducing the chance of missing an appropriate direction, a second idea proposed by Harris and Stephens [Harris and Stephens: 1988] to combat the non-isotropic shortcoming of Moravec's algorithm. Now with an analytic expansion the direction (ξ,η) need not be restricted to four discrete directions, so that we can explore all directions. Let θ represent the direction of (ξ,η) as represented by the unit vector $T = (\cos(\theta),\sin(\theta))$, so that we can rewrite

$$J(x,y;\theta) = \begin{bmatrix} \cos(\theta) & \sin(\theta) \end{bmatrix} M \begin{bmatrix} \cos(\theta) \\ \sin(\theta) \end{bmatrix}$$

Thus,

$$\begin{aligned} \overline{J}(x,y) &= \min_{\theta} J(x,y;\theta), \\ &= \min(\lambda_1,\lambda_2). \end{aligned}$$

where λ_1,λ_2 are the eigenvalues of the matrix M. Note that $\lambda_1 = \lambda_1(x,y)$ and $\lambda_2 = \lambda_2(x,y)$ can vary spatially. Finally, the interest points are the local max of $\overline{J}(x,y)$.

The Moravec/Harris-Stephens corner detection then works by first computing the matrix M. This is done by computing a numerical derivative, central difference mask $(-1,0,1)$ in Harris-Stephens, and $(-2,-1,0,1,2)$ in Schmid et al [Schmid: et al]. In other cases, the Sobel operator is used to estimate derivatives. A Gaussian derivative has also been used to estimate derivatives and this has been shown to improve results [Schmid:IJCV2000]. The second step is smoothing M with a Gaussian mask ($\sigma = 2$ in Schmid). Note that since we are

computing a nonlinear term, *eg.*, f_x^2, it is not necessarily the case that the filter commutes with the square operation and generally we expect,

$$G * f_x^2 \neq (G * f_x)^2$$

Second, instead of computing the eigenvalues, the algorithm computes the trace(Tr) and determinant(Det) of the matrix M using

$$\lambda_{1,2} = \frac{Tr \pm \sqrt{Tr^2 - 4Det}}{2}.$$

Since not all local maxima of $\lambda_{1,2}(x,y) = \min(\lambda_1(x,y), \lambda_2(x,y))$ are of interest due to noise, we need to threshold very small λ_{min}. This needs to be done with care since interest detection needs to be stable with variations in illumination: a scaling of $f(x,y)$ by α scale λ_{min} by α^2. Thus, a judgement of interest point needs to be invariant to such scaling. Thus, we need to first normalize λ_{min} and λ_{max} to $\left(\frac{\lambda_{min}}{\lambda_{max}}, 1\right)$, i.e., The λ_1, λ_2 space is divided into an equivalence class of lines, parameterized by the angle of that line. Thus we have two conditions:

$$\begin{cases} \lambda_{min}^2 + \lambda_{max}^2 & > & \tau_1 \\ \frac{\lambda_{min}}{\lambda_{max}} & > & \tau_2 \end{cases}$$

A single function $det - \alpha Tr^2$ captures the intersection of these regions, Figure 2(e), where α and R_0 replace τ_1 and τ_2. Observe that the matrix M intrinsically involves a socle parameter associated with h, namely the detection scale σ_h. In addition, in computing the derivatives of f, f_x and f_y, it is wise to use a Gaussian pre-filter which has a differentiation scale σ_d. While we expect σ_d to be 0.7 or 1, the scale of the detector σ_h can vary quite widely, depending on the scale of expected structures in the image. The approach of Lindeberg [Reference] can be adopted to scale selection by requiring the corner strength function R is maximum over both space and scale σ_h [Mikolajczyk and Schmid, IJCV 04]

1.1 The Corner Detection Algorithm

1. Filter the image $f(x,y)$ with spatial derivatives of a Gaussian $G(x,y,\sigma_1)$, where σ_1 is called the differentiation scale, typically $\sigma_1 = 0.7$ or 1, *i.e.* G_x and G_y are used to estimate the gradient of f:

$$\begin{cases} f_x & = & G_x * f \\ f_y & = & G_y * f \end{cases}$$

2. Form three spatial maps:

$$\begin{cases} A(x,y) & = & (G_x * f)^2 \\ B(x,y) & = & (G_x * f)(G_y * f) \\ C(x,y) & = & (G_y * f)^2 \end{cases}$$

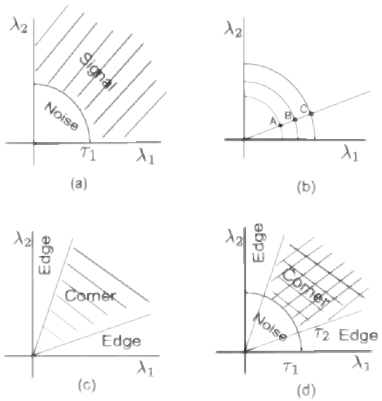


Figure 1: (a) In addition, For certain values of λ below a threshold, we may be dealing with noise. (b) Points A,B,C which can arise from the same structure under illumination changes must be considered equivalent. (c) An edge is determined by $\frac{\lambda_{min}}{\lambda_{max}} < \tau_2$ which translates into a central beam (d) Both conditions $\lambda_1^2 + \lambda_2^2 > \tau_1$ and $\frac{\lambda_{min}}{\lambda_{max}} < \tau_2$. A single function $R_\alpha(x,y) = det - \alpha Tr^2 = R_0$ separates the domain into edge or noise $R(x,y) < R_0$ and corner $R(x,y) > R_0$. (e) variations of $R_\alpha(x,y) = R_0$ with R_0 shows that R_0 controls the extent of noise.

3. Blur these maps with a Gaussian $G(x,y,\sigma_2)$, where σ_2 is the integration scale which can vary over a range of scales, *e.g.* $\sigma_2 = 2.0$ or 3.0, *etc.*:

$$\begin{cases} \overline{A}(x,y) &= G(x,y,\sigma_2) * A \\ \overline{B}(x,y) &= G(x,y,\sigma_2) * B \\ \overline{C}(x,y) &= G(x,y,\sigma_2) * C \end{cases}$$

4. Compute

$$\begin{aligned} tr(x,y) &= \overline{A}(x,y) + \overline{C}(x,y) \\ det(x,y) &= \overline{AC} - \overline{B}^2 \end{aligned}$$

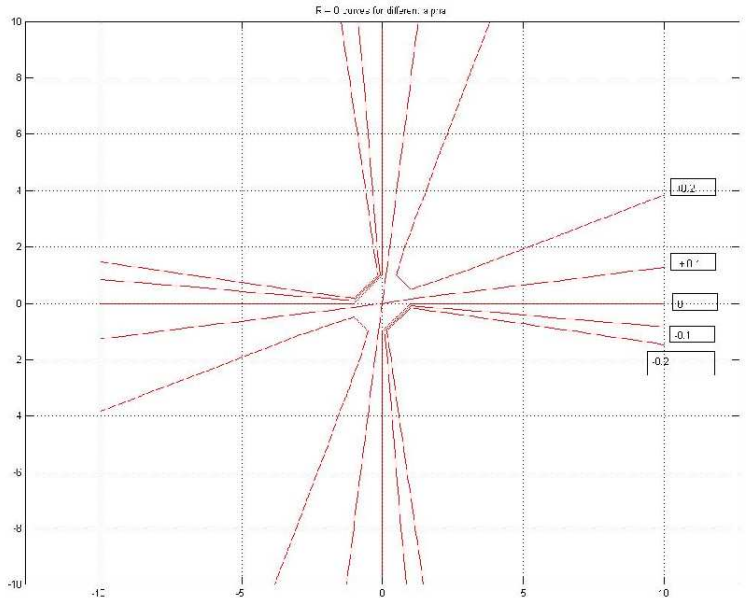
5. Compute

$$R(x,y) = det(x,y) - \alpha\ tr^2(x,y), \tag{2}$$

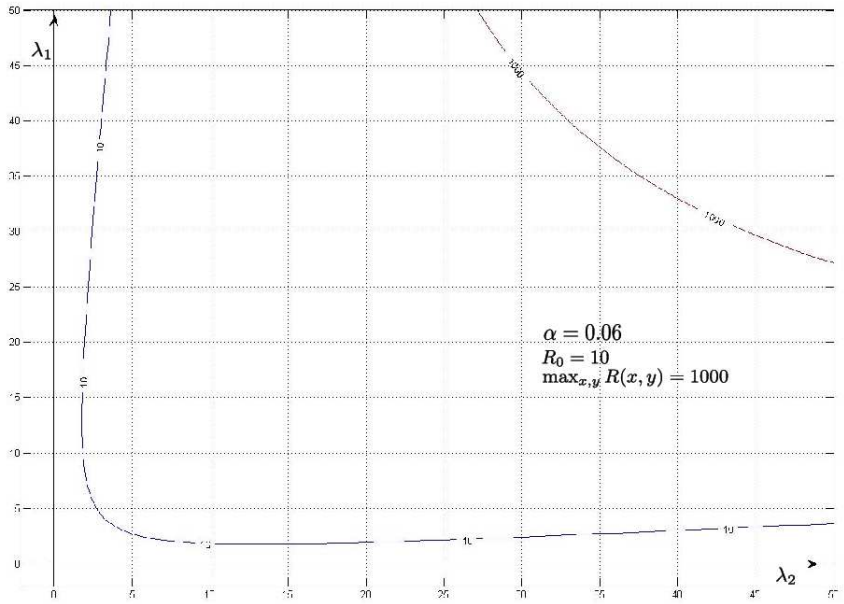
where α is a parameter of the system

6. Compute local max of $R(x,y)$ by applying non-maximum suppression over a 3×3 neighborhood of each point.

7. Threshold R_0 is used to prune points with $R(x,y) < R_0$, where $R_0 = 0.01 \times \max_{x,y} R(x,y)$



(f)



(g)

Figure 2: (f) variations of $R_\alpha(x, y) = R_0$ with α shows that α controls the edge/corner selection. (g) A typical choice is $\alpha = 0.06$ and $R_0 = 0.01 \times \max_{x,y} R(x, y)$.

8. Return all points (x, y) with strength $R(x, y)$.

1.2 Motivating matrix M using applications

Tomasi and Kanade [Tomasi: Kanade: CMU91] argue that features should not be arbitrarily defined but rather be considered in the context of how they may be used in applications, the tracking problem in their case. The tracking problem is approached by defining a local model for the spatiotemporal image $f(P, t)$, where P is location (x, y) and t is time, namely, that

$$f(P, \tau + \Delta\tau) = f(P - \Delta P, \tau) + \eta(P)$$

where $\eta(P)$ is noise. The problem becomes one of estimating ΔP by minimizing

$$E(\overline{P}, \tau, \Delta P, \Delta\tau) = \int \int_W [f(P - \Delta P, \tau) - f(P, \tau + \Delta\tau)]^2 \, h(\overline{P} - P) dP$$

where $h(P)$ is a weighting function, *e.g.*, a Gaussian. Using a first-order approximation, we can write

$$f(P - \Delta P, \tau) = f(P, \tau) - \nabla f^T(P, \tau) \Delta P,$$

so that we have to minimize

$$E(\overline{P}, \tau, \Delta P, \Delta\tau) = \int \int_W \left[f(P, \tau) - f(P, \tau + \Delta\tau) - \nabla f^T(P, \tau) \Delta P\right]^2 h(\overline{P} - P) dP,$$

which by differentiation gives a vector equation

$$\int \int_W \left[f(P, \tau) - f(P, \tau + \Delta\tau) - \nabla f^T(P, \tau) \Delta P\right] \nabla f(P, \tau) h(\overline{P} - P) dP = 0.$$

Using $\left(\nabla f^T \Delta P\right) \nabla f = (\nabla f \nabla f^T) \Delta P$,

$$\left[\int \int_W (\nabla f \nabla f^T) h(\overline{P} - P) dP\right] \Delta P = \int \int_W [f(P, \tau) - f(P, \tau + \Delta\tau)] \nabla f(P, \tau) h(\overline{P} - P) dP.$$

Observe that

$$\nabla f \; \nabla f^T = \begin{bmatrix} f_x \\ f_y \end{bmatrix} \begin{bmatrix} f_x & f_y \end{bmatrix} = \begin{bmatrix} f_x^2 & f_x f_y \\ f_x f_y & f_y^2 \end{bmatrix},$$

and the left-hand side can be written as $[(\nabla f \nabla f^T) * h]\Delta P$, which using

$$M = \begin{bmatrix} h * f_x^2 & h * f_x f_y \\ h * f_x f_y & h * f_y^2 \end{bmatrix},$$

gives $M\Delta P$. Similarly, the right-hand side can be written as

$$\begin{aligned} b(\overline{P}) &= \int\int_W [f(P,\tau) - f(P,\tau+\Delta\tau)]\,\nabla f(P,\tau) h(\overline{P}-P)dp, \\ &= [[f(P,\tau) - f(P,\tau+\Delta\tau)]\,\nabla f(P,\tau)] * h(P) \end{aligned}$$

Now, $[f(P,\tau) - f(P,\tau+\Delta\tau)]$ is just the frame difference, and ∇f is the spatial image gradient, both readily computable. This gives the basic equation for tracking.

$$M \cdot \Delta P = b \tag{3}$$

The problem with Equation (3) is that when the intensity is roughly consistent the matrix M is null and Equation (3) does not provide sufficient information. Rather Equation (3) can best be solved at specific locations in the image, hence the idea of feature points! Features are locations at which Equation (3) can be solved well and this requires that

1. coefficients of M be well above image noise level, which implies that both eigenvalues must exceed some threshold τ_1, and

2. The Matrix M is well-conditioned, *i.e.*, The ratio of eigenvalues is bounded,

$$\tau_2 < \left|\frac{\lambda_1}{\lambda_2}\right| < \frac{1}{\tau_2} \text{ or } \frac{\lambda_{min}}{\lambda_{max}} > \tau_2$$

Tomasi and Kanade then claim that "In practice, when the smaller eigenvalue is sufficiently large to meet the noise criteria, the matrix M is usually also well conditioned. Thus, due to the fact that the intensity variations in a window are bounded by the maximum allowable pixel value, so that the greater eigenvalue cannot be arbitrarily large". Thus, a feature simply requires the first condition, which can be written as

$$\lambda_{min} > \tau_1$$

where $\lambda_{min} = \min(\lambda_1, \lambda_2)$.

1.3 Results

Schmid et. al. compares the performance of various corner detectors under various transformations. Specifically two corners are considered equivalent if they are related by the

expected transformation., *i.e.*, they lie within an ϵ distance from each other. They measure repeatability of detected corners over image rotations, Figure 3(a), changes in scale Figure 3(b), changes in illumination, both uniform, Figure 3(c) and non-uniform, Figure 3(d), viewpoint changes, Figure 3(e) and camera noise. It is clear that corners are stable with all but scale changes and viewpoint variations.

2 Automatic Scale Selection

Consider Figure 4 which depicts objects (flowers) occurring at a range of sizes. clearly, any feature detector which has a fixed scale of operation would only respond to a narrow band of features Figure 5. On the other hand, operating the detector at all relevant scales would lead to a large number of spurious features. The idea arises whether the response of the detector itself can be used to determine the appropriate scale: in matched filtering, the highest response filter would give the best indication of the signal. Thus, maximizing the response of a feature detector over all scales would presumably give the best intrinsic scale of the signal. However maximizing response over scales requires a comparison of the response of the operator at two separate scales which requires a calibration of responses across scales. In order to calibrate the response of the detector at one scale to another, we use a common point of reference, an object scaled in size with the same proportion.

Feature detectors are typically local and can often be written in terms of the intensity function and its derivatives, $f, f_x, f_y, \ldots$. We assume that the operator scale is defined by the width if the Gaussian used to generate it.

In general, a Gaussian scale space is generated by smoothing the signal with a continuum of scales:

$$\overline{f}(x, y, \sigma) = G(x, y, \sigma) * f(x, y)$$

where

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right).$$

Observe that the Gaussian kernel $G(x, y, t)$ is the Green's function for the Diffusion equation,

$$u_t = \frac{1}{2}\Delta u,$$

where

$$G(x, y, t) = \frac{1}{2\pi t} \exp\left(-\frac{x^2 + y^2}{2t}\right).$$

We take $t = \sigma^2$ so that $\frac{dt}{d\sigma} = 2\sigma$ leading to $u_\sigma = 2\sigma u_t$. Thus the heat equation can be written as,

$$u_\sigma = \sigma\Delta u,$$

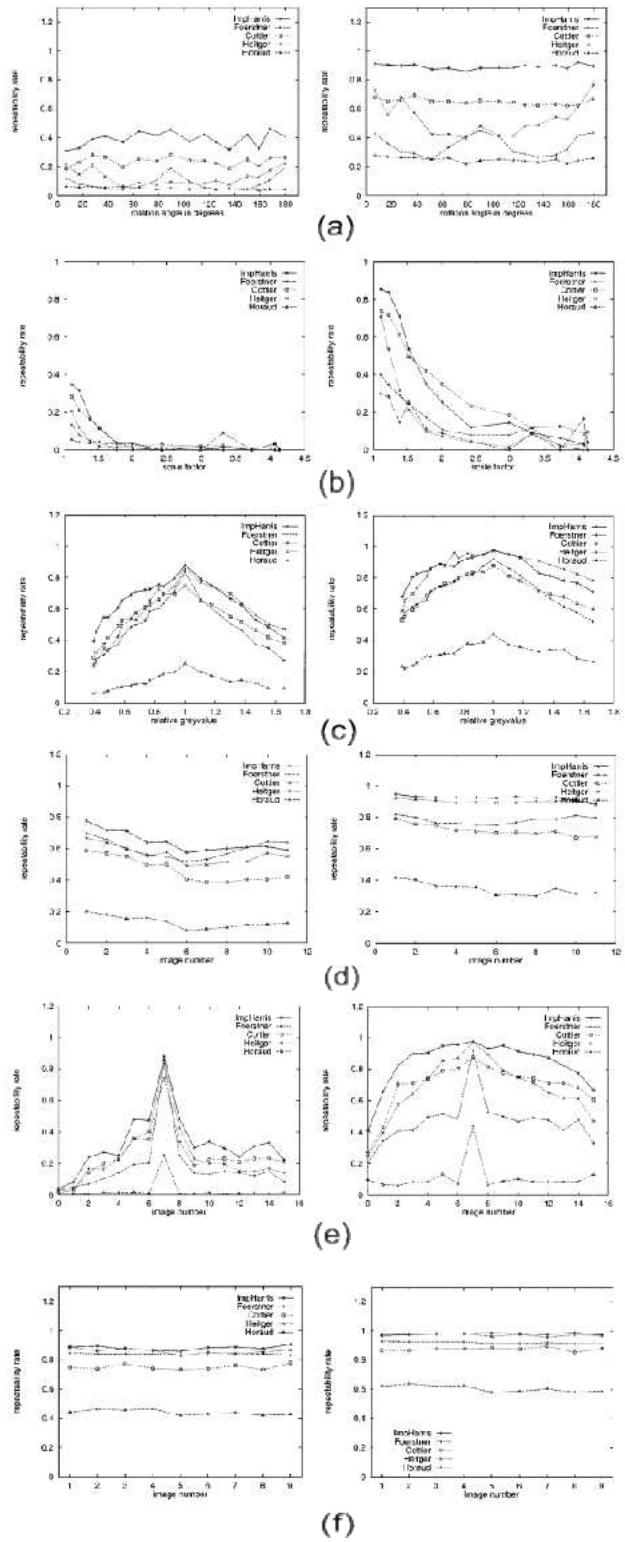


Figure 3: Repeatability rate with $\epsilon = 0.5$(left column) and $\epsilon = 1.5$(right column) for (a) rotation (b) scale changes (c) uniform illumination changes (d) non-uniform illumination changes (e) viewpoint change and (f) camera noise.

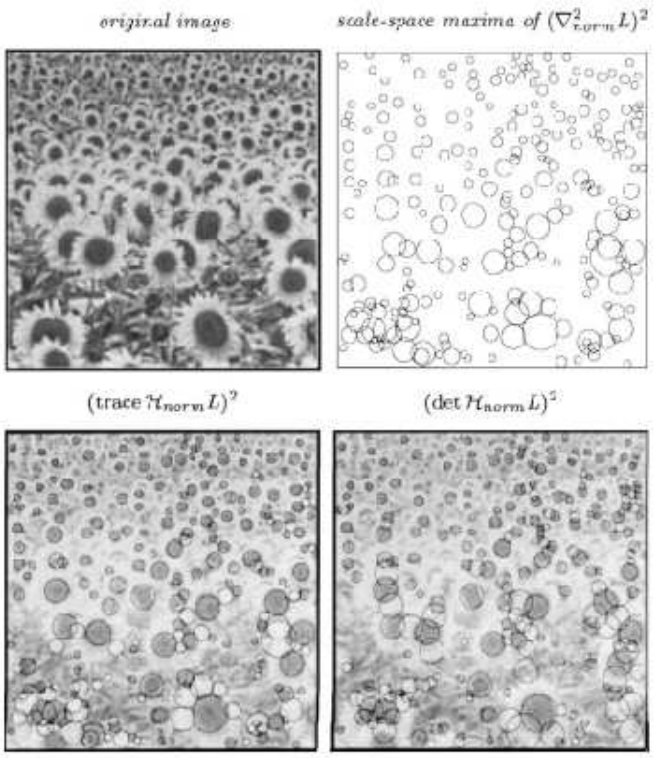


Figure 4: From [Lindeberg:IJCV98]: Normalized scale-space maxima computed from an image of a sunflower field: (top left): Original image. (top right): Circles representing the 250 normalized scale-space maxima of $(traceH_{norm}L)^2$ having the strongest normalized response. (bottom left): Circles representing scale-space maxima of $(traceH_{norm}L)^2$ superimposed onto a bright copy of the original image. (bottom right): Corresponding results for scale-space maxima of $(detH_{norm}L)^2$.

for which $G(x, y, \sigma)$ is the Green's function, itself satisfying this equation

$$G_\sigma = \sigma \Delta G.$$

Suppose now that $f(x, y)$ depicts an object (*e.g.* a flower) of scale σ_0 at (x_0, y_0). Suppose that $g(x, y)$ is the image of the same object at half the distance so that $g(x, y) = f(\frac{x}{2}, \frac{y}{2})$. In general, if the object is scaled up by a factor λ we have,

$$g(x, y) = f\left(\frac{x}{\lambda}, \frac{y}{\lambda}\right),$$

where the scaled object now has the scale of $\lambda\sigma_0$. We can now explore the scale space of f and g in relation to each other:

Proposition 2.1. *Consider a function $f(x, y)$ scaled as,*

$$g(x, y) = f\left(\frac{x}{\lambda}, \frac{y}{\lambda}\right).$$

Then the Gaussian scale space of g, $\overline{g}(x, y)$ is a scaled version of the scale space of f, in space and scale, but it is attenuated by $\frac{1}{\lambda^2}$:

$$\lambda^2 \overline{g}(\lambda x, \lambda y, \lambda\sigma) = \overline{f}(x, y, \sigma).$$

This can also be written as,

$$\sigma_2^2 \overline{g}\left(\frac{\sigma_2}{\sigma_1}x, \frac{\sigma_2}{\sigma_1}y, \sigma_2\right) = \sigma_1^2 \overline{f}(x, y, \sigma_1).$$

Proof. The Gaussian scale space of g is defined as

$$\begin{aligned} \overline{g}(\overline{x}, \overline{y}, \overline{\sigma}) &= G(\overline{g}(\overline{x}, \overline{y}, \overline{\sigma}) * g(\overline{x}, \overline{y}) \\ &= G(\overline{g}(\overline{x}, \overline{y}, \overline{\sigma}) * f\left(\frac{\overline{x}}{\lambda}, \frac{\overline{y}}{\lambda}\right) \end{aligned}$$

and

$$\overline{f}(x, y, \sigma) = G(x, y, \sigma) * f(x, y).$$

We can see that, using $\overline{x} = \lambda x, \overline{y} = \lambda y$

$$\overline{g}(\lambda x, \lambda y, \overline{\sigma}) = G(\lambda x, \lambda y, \overline{\sigma}) * f(x, y),$$

and since

$$\begin{aligned} G(\lambda x, \lambda y, \overline{\sigma}) &= \frac{1}{2\pi\overline{\sigma}^2} \exp\left(-\frac{\lambda^2(x^2 + y^2)}{2\sigma^2}\right) \\ &= \frac{\frac{1}{\lambda^2}}{2\pi\left(\frac{\overline{\sigma}}{\lambda}\right)^2} \exp\left(\frac{x^2 + y^2}{2\left(\frac{\overline{\sigma}}{\lambda}\right)^2}\right) \end{aligned}$$

Using $\overline{\sigma} = \lambda\sigma$, we have

$$\begin{aligned} \overline{g}(\lambda x, \lambda y, \lambda\sigma) &= \frac{1}{\lambda^2} G(x, y, \sigma) * f(x, y) \\ &= \frac{1}{\lambda^2} \overline{f}(x, y, \sigma) \end{aligned}$$

□

In other words, is f is scaled up by a factor of 3, then the scale-space is also scaled up by a factor of 3, both spatially and in scale, and it is attenuated by a factor of 9. This tells us that comparing a detector response at scales 1 and 3 dreictly would be comparing apples and oranges, since if the object is truely at scale 3, then the response of the operator at scale 1 represents an unfair $9:1$ advantage! Rather, in order to compare detector responses at scale 1 and 3, we must

1. compute the detector with derivatives at the respective scales and
2. scale up the response by $\lambda^2 = 9$.

This can be done by multiplying each scale by its own scale:

$$(\lambda\sigma)^2\overline{g}(\lambda x, \lambda y, \lambda\sigma) = \sigma^2\overline{f}(x, y, \sigma).$$

the net effect of the above two factors is as follows. First, the spatial derivatives of g and f are related as

$$\begin{cases} \frac{\partial g}{\partial \overline{x}} &= \frac{1}{\lambda}\frac{\partial f}{\partial x}\left(\frac{x}{\lambda}, \frac{y}{\lambda}\right) \\ \frac{\partial g}{\partial \overline{y}} &= \frac{1}{\lambda}\frac{\partial f}{\partial y}\left(\frac{x}{\lambda}, \frac{y}{\lambda}\right) \end{cases}$$

and

$$\begin{cases} \frac{\partial^2 g}{\partial \overline{x}^2} &= \frac{1}{\lambda^2} f_{xx} \\ \frac{\partial^2 g}{\partial \overline{x}\overline{y}} &= \frac{1}{\lambda^2} f_{xy} \\ \frac{\partial^2 g}{\partial \overline{y}^2} &= \frac{1}{\lambda^2} f_{yy} \end{cases}$$

which can be written as,

$$\begin{cases} (\lambda\sigma) g_{\overline{x}} &= \sigma f_x \\ (\lambda\sigma) g_{\overline{y}} &= \sigma f_y \end{cases}$$

and

$$\begin{cases} (\lambda^2\sigma^2) g_{\overline{x}\overline{x}} &= \sigma^2 f_{xx} \\ (\lambda^2\sigma^2) g_{\overline{x}\overline{y}} &= \sigma^2 f_{xy} \\ (\lambda^2\sigma^2) g_{\overline{y}\overline{y}} &= \sigma^2 f_{yy} \end{cases}$$

and so on for higher derivatives. In general

$$\lambda^n \frac{d^m}{d\overline{x}^m}\overline{g} = \frac{d^m \overline{f}}{dx^m}$$

That spatial derivatives differ by a factor λ^m gives rise to the notion of a Normalized derivatives [Lindeberg:IJCV98], where each derivative is multiplied by σ, *i.e.*, in computing a detector response whenever f_x, f_y are used, these are replaced by $(\sigma f_x, \sigma f_y)$, and similarly

$(\sigma^2 f_{xx}, \sigma^2 f_{xy}, \sigma^2 f_{yy})$ in place of (f_{xx}, f_{xy}, f_{yy}), *etc.* while in each fixed scale. This simply scales each by a constant, in the scale space if scales each scale plane appropriately. The use of normalized derivatives allows for a calibration of local differential operators across scales. This property ensures that as an object scales by a factor of λ, then a spatial maximum in space and scale of (x_0, y_0, σ_0) in the unscaled case would also lead to a spatial maximum in space and scale at $(\lambda x_0, \lambda y_0, \lambda \sigma_0)$.

The idea of scale selection in the scale space goes back to Crowley [Crowley:PhD:1981, Crowley:Parker:PAMI:1984] who picked the sufficiently strong local extrema in the DoG pyramid, a precursor to the modern methods using differential operators. Lindeberg [Lindeberg:IJCV:1998] uses the Laplacian of Gaussian (LOG) to find the local extrema in space and scale of scale-normalized differential operators. He showed that the normalization of Laplacian operator by the square of the scale of the Gaussian, σ^2 leads to scale-invariant measurements [Lindeberg: journal of applied statistics 1994].

3 SIFT

Lowe [Lowe:IJCV04] points out that a Difference of Gaussians operator when taken at multiplicative scales of σ and $k\sigma$ has the scale invariant property required by Lindeberg's condition:

$$\begin{aligned} G(x, y, k\sigma) - G(x, y, \sigma) &\simeq (k\sigma - \sigma)\frac{\partial G}{\partial \sigma}(x, y, \sigma) && (4) \\ &= (k-1)\sigma\sigma\Delta G && (5) \\ &= (k-1)\sigma^2\Delta G && (6) \end{aligned}$$

A typical value for K is $\sqrt{2}$, see Figure 6 . Each octave (doubling of σ) is divided into S intervals so that $K = 2^{\frac{1}{S}}$ and $S + 3$ images are produced for each octave.

The 3-D extrema of $D(x, y, \sigma)$

$$\begin{aligned} D(x, y, \sigma) &= [G(x, y, k\sigma) - G(x, y, \sigma)] * f(x, y) \\ &= \overline{f}(x, y, k\sigma) - \overline{f}(x, y, \sigma), \end{aligned}$$

are obtained by checking a $3 \times 3 \times 3$ neighborhood in x,y and σ, Figure . Check if the central pixel is the largest or smallest of these 27 pixels, as in Figure **??**.

The frequency of sampling in scale was experimentally shown to be 3 per octave, frequency of sampling in space was fairly arbitrary and $\sigma = 1.6$ was selected. The image resolution is doubled prior to building the first level of the pyramid.

3.1 Subpixel Localization

Once a keypoint has been detected at a pixel and scale (x_0, y_0, σ_0), its location and scale can be estimated to subpixel accuracy using a second order Taylor approximation of $D(x_0, y_0, \sigma_0)$

with respect to the discrete pixel location (x_0, y_0),

$$D(x_0+\Delta x, y_0+\Delta_y, \sigma_0+\Delta_\sigma) = D(x_0, y_0, \sigma_0)+\nabla D(x_0, y_0, \sigma_0) \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\sigma \end{bmatrix} + \begin{bmatrix} \Delta x & \Delta y & \Delta\sigma \end{bmatrix} H_D(x_0, y_0, \sigma_0) \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta\sigma \end{bmatrix}$$

where H_D is the Hessian of D. Setting $\nabla D = 0$ gives,

$$\nabla D(x_0, y_0, \sigma_0) = H_D(x_0, y_0, \sigma_0) \begin{bmatrix} \Delta x^* \\ \Delta y^* \\ \Delta\sigma^* \end{bmatrix}$$

giving the offset of the extrema from the current pixel location as,

$$\begin{bmatrix} \Delta x^* \\ \Delta y^* \\ \Delta\sigma^* \end{bmatrix} = H_D^{-1}(x_0, y_0, \sigma_0)\nabla D(x_0, y_0, \sigma_0)$$

giving a final location and scale $(x + \Delta x^*, y + \Delta y^*, \sigma + \Delta\sigma^*)$. Then,

$$D(x + \Delta x^*, y + \Delta y^*, \sigma + \Delta\sigma^*) = D(x_0, y_0, \sigma_0) + \frac{1}{2}\nabla D^T H_D \nabla D$$

All extrema with a value of $D((x + \Delta x^*, y + \Delta y^*, \sigma + \Delta\sigma^*)$ less then 0.03 of the total range was rejected. However, peaks which are along edges can give strong D value, even if unstable to noise. These points can only be rejected by consulting the Hessian of d which indicates the curvature of the function D in various directions which correlates with stability,

$$H = \begin{bmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{bmatrix}$$

Observe that using $Tr(H) = \lambda_1 + \lambda_2$ and $det(H) = \lambda_1\lambda_2$, where λ_1, λ_2 are the eigenvalues of H, we have,

$$\frac{Tr^2(H)}{Det(H)} = \frac{(\lambda_1 + \lambda_2)^2}{\lambda_1\lambda_2} = \frac{\left(\frac{\lambda_1}{\lambda_2} + 1\right)^2}{\frac{\lambda_1}{\lambda_2}}$$

which in analogy to Harris-Stephans treatment focuses on theratio of eigenvalues, cutting off those with ratios between 0.1 and 10. An example of SIFT feature points on an image is shown in Figure 7

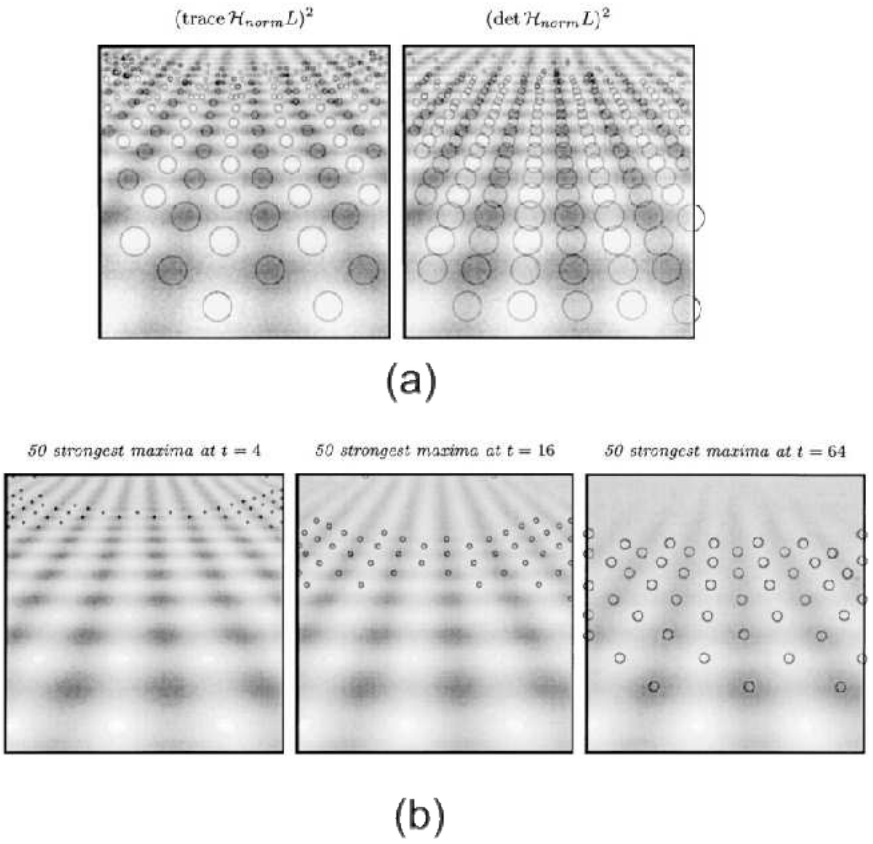


Figure 5: From [Lindeberg:IJCV98] (a) The 250 most significant normalized scale-space extrema detected from the perspective projection of a sine wave (with 10% added Gaussian noise). (b) The 50 strongest spatial responses to the Laplacian operator computed at the scale levels: (a) t = 4.0, (b) t = 16.0, and (c) t = 64.0. Observe how this blob detector leads to a bias towards image structures of a certain size.

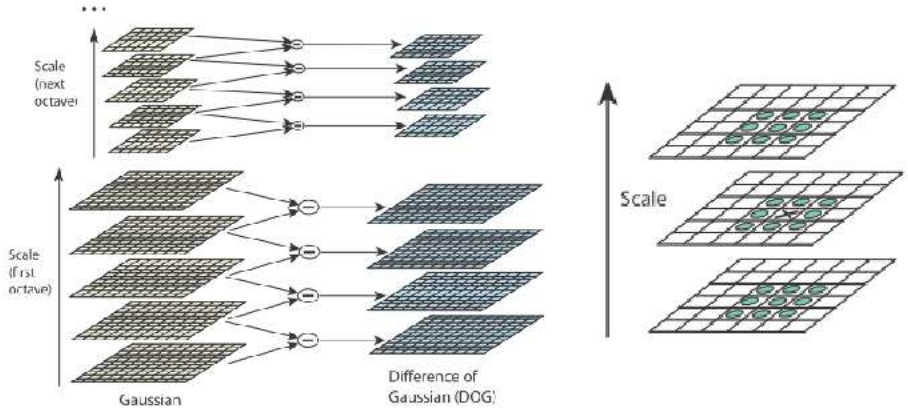


Figure 6: From Lowe: (left) For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space images shown on the left. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images on the right. After each octave, the Gaussian image is down-sampled by a factor of 2, and the process repeated. (right) Maxima and minima of the difference-of-Gaussian images are detected by comparing a pixel (marked with X) to its 26 neighbors in 3x3 regions at the current and adjacent scales (marked with circles)

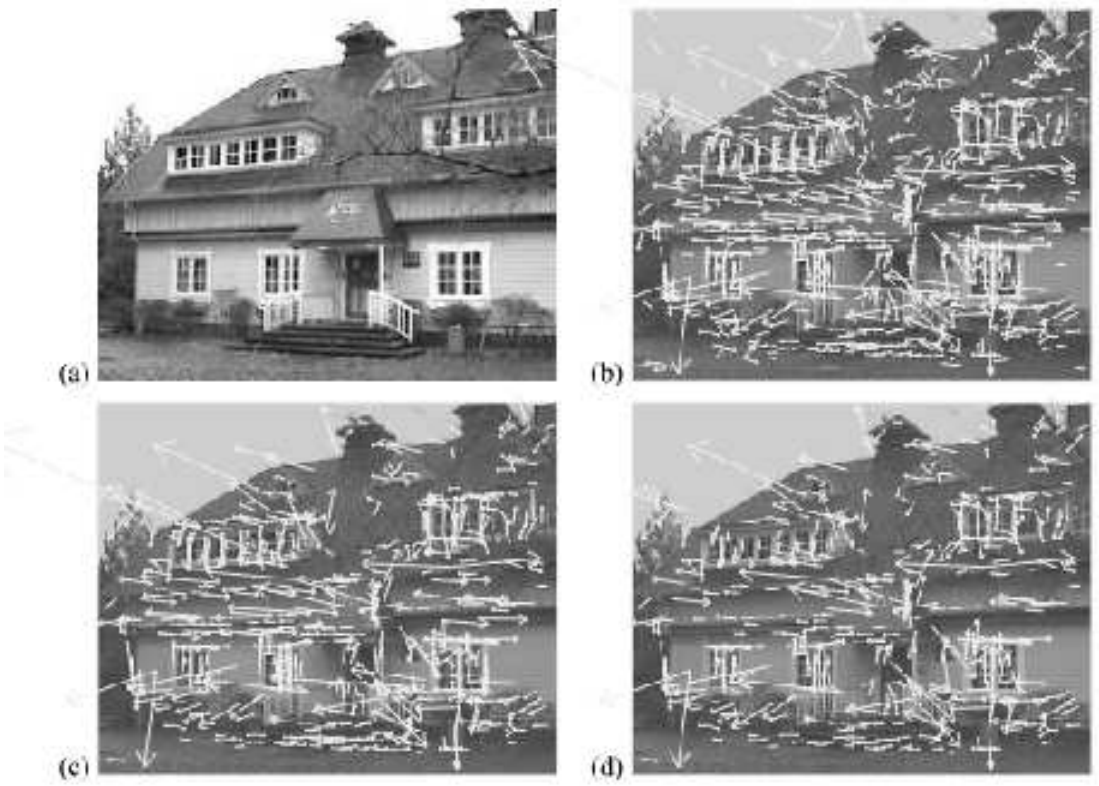


Figure 7: From Lowe: This Figure shows the stages of keypoint selection. (a) The 233x189 pixel original image. (b) The initial 832 keypoints locations at maxima and minima of the difference-of-Gaussian function. Keypoints are displayed as vectors indicating scale, orientation, and location. (c) After applying a threshold on minimum contrast, 729 keypoints remain. (d) The final 536 keypoints that remain following an additional threshold on ratio of principal curvatures