

Super Resolution for Automated Target Recognition

Maksim Levental

Abstract—Super resolution is the process of producing high-resolution images from low-resolution images while preserving ground truth about the subject matter of the images and potentially inferring more such truth. Algorithms that successfully carry out such a process are broadly useful in all circumstances where high-resolution imagery is either difficult or impossible to obtain. In particular we look towards super resolving images collected using longwave infrared cameras since high resolution sensors for such cameras do not currently exist. We present an exposition of motivations and concepts of super resolution in general and current techniques, with a qualitative comparison of such techniques. Finally we suggest directions for future research.

1 IMAGE REGISTRATION

1.1 Global Algorithms	1
1.1.1 Feature Detection and Selection	1
1.1.1.1 Harris Corner Detection	1
1.1.1.2 SIFT	3
1.1.2 Feature Matching	3
1.1.2.1 Normalized Cross-correlation	3
1.1.3 Mutual Information	5
1.1.4 Transform Estimation	6
1.2 Gaussian Process	6

Images obtained from multiple vantage points, or at different times, of the same scene, become distorted with respect to each other. Since in MISR the aim is to exploit new information across multiple LR samples we need to first rectify these distortions and reconcile the images. Effectively this means finding one or more pixel transformations that enable mapping all LR images to a common pixel grid. When the transformations cannot be deduced from first principles (e.g. precise knowledge of the relative motion of the scene and the imaging system) they must be estimated from the LR images. There are broadly two perspectives on constructing the transformation f (see eqn. (??)): the global perspective which aims to model motion as a map of the image as a whole and the local perspective which aims to model motion as a deformation of individual pixels. These two perspectives naturally correspond to a globally or locally defined transformation f . We first cover global algorithms, examining the different techniques available for each step, and then move on to local algorithms.

1.1 Global Algorithms

Most global image registration algorithms consist of a feature detection and selection step (also called Control Point (CP) selection), a feature matching step, and a transform estimation step.

1.1.1 Feature Detection and Selection

Feature detection and selection is the process of identifying features of the image that are presumed to be invariant across the multiple images to be registered. Note that here by features we mean image artifacts (e.g. edges, contours, line intersections, or corners); in this context encodings or transformations of these image artifacts are called *descriptors*. The CPs are the data that will be used to estimate the transformation f . Therefore, in order that the estimated transformation is accurate, CPs should be robust to noise and image degradation, sufficiently distributed throughout the image, and readily matched in the matching step.

1.1.1.1 Harris Corner Detection

Bentoutou *et al.*[1] use a Harris detector[2] to find corner points, arguing that corners are robust to noise and stable over multiple images. The Harris detector improves on the Moravec[3] detector. The Moravec detector starts from the error function $E_{x,y}(u, v)$ which computes the sum of the squared differences (SSD) between an $m \times m$ weighted window around a pixel $X(x, y)$ and weighted windows shifted by u, v pixels:

$$E_{x,y}(u, v) := \sum_{i,j=-m/2}^{m/2} w_{ij} [X(x_i + u, y_j + v) - X(x_i, y_j)]^2 \quad (1)$$

where $x_i := x + i$ and $y_j := y + j$. Moravec assigns a "corner score" according to the following reasoning (see figure 2):

- 1) If a pixel is in a region of uniform intensity then $E_{x,y}(u, v)$ is small for all u, v (since neighboring windows are similar).
- 2) If a pixel is on an edge, then $E_{x,y}(u, v)$ for either $u > 0$ or $v > 0$, but not both, is high.
- 3) If a pixel is on a corner, then $E_{x,y}(u, v)$ for $u > 0$ and $v > 0$ is high.

Therefore the corner score at pixel coordinate (x, y) is $\min_{u,v} E_{x,y}(u, v)$ in order to select for the third case. Moravec comments that this corner score is not isotropic, i.e. if edges aren't aligned with either the pixel axes or diagonals then $E_{x,y}(u, v)$ will incorrectly be low. Harris' insight was to linearize $E_{x,y}(u, v)$ in order to compute a quantity more closely related to the intensity variation in a local neighborhood of a pixel:

$$X(x_i + u, y_j + v) \approx X(x_i, y_j) + \frac{\partial X}{\partial u} u + \frac{\partial X}{\partial v} v \quad (2)$$

where the partial derivatives are taken at (x, y) . This implies

$$E_{x,y}(u, v) \approx \sum_{i,j=-m/2}^{m/2} w_{ij} [X_u u + X_v v]^2 \quad (3)$$

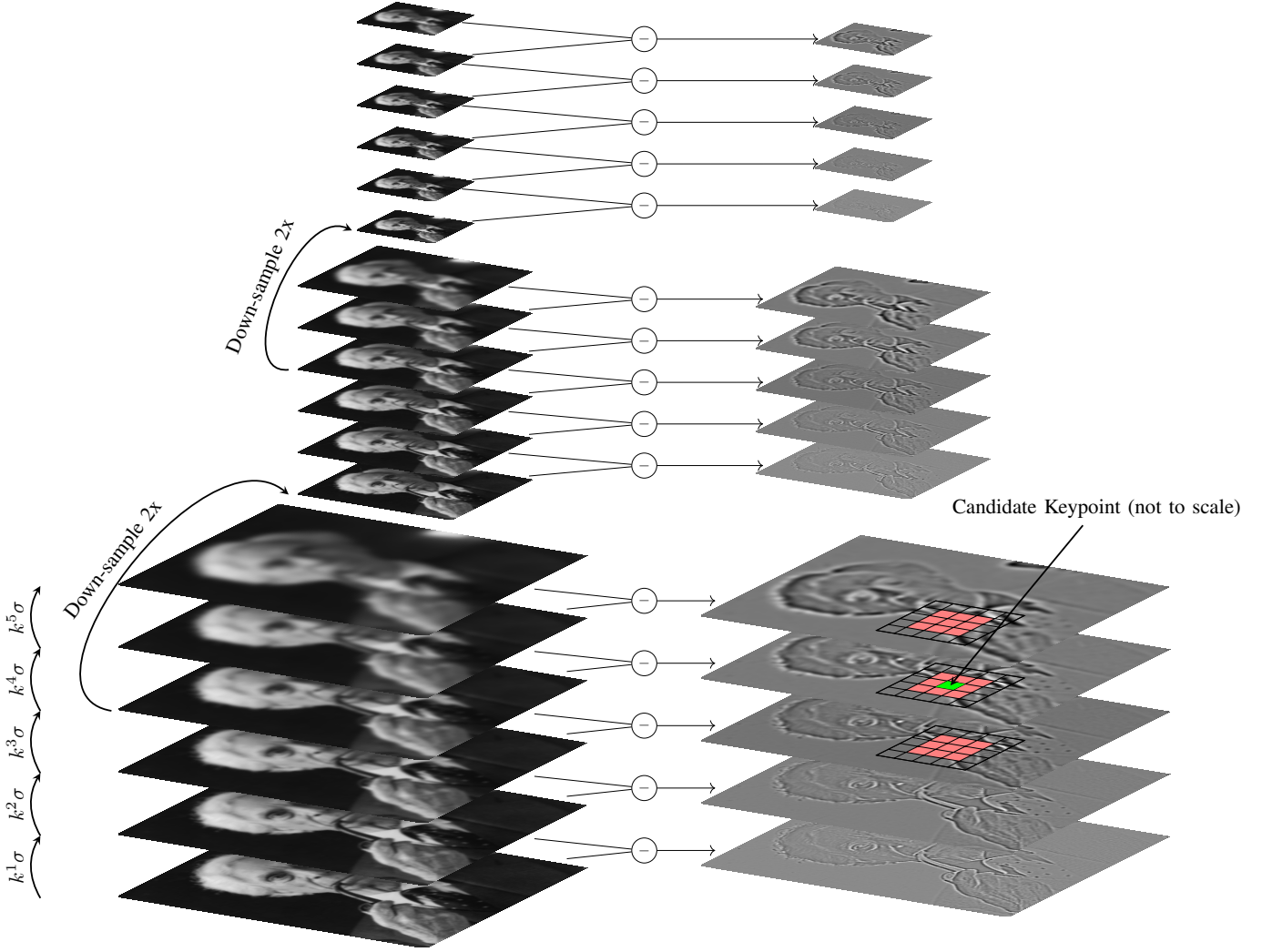


Fig. 1: Gaussian filtered and sub-sampled pyramid and corresponding difference of Gaussians pyramid for identifying keypoints (as local gradient maxima).

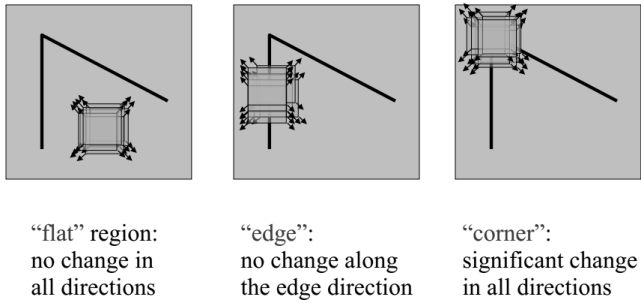


Fig. 2: Moravec Corner Detector

$$= \sum_{i,j=-m/2}^{m/2} w_{ij} [X_u^2 u^2 + X_v^2 v^2 + 2X_u X_v uv] \quad (4)$$

$$= [u, v] \begin{bmatrix} \sum w_{ij} X_u^2 & \sum w_{ij} X_u X_v \\ \sum w_{ij} X_u X_v & \sum w_{ij} X_v^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} \quad (5)$$

$$= [u, v] M \begin{bmatrix} u \\ v \end{bmatrix} \quad (6)$$

where $X_u = \partial X / \partial u$ and similarly X_v . The matrix in eqn. (6), called the *structure tensor* or *second-moment matrix* M , is the quantity Harris investigated. Harris reasoned that the cases of Moravec correspond to conditions on the eigenvalues λ_1, λ_2 of M :

- 1) If $\lambda_1 \approx \lambda_2 \approx 0$ then $X(x, y)$ is in a region of uniform intensity.
- 2) If $\lambda_1 \gg \lambda_2$ or $\lambda_2 \gg \lambda_1$ then $X(x, y)$ is on an edge.
- 3) $\lambda_1 \approx \lambda_2 > 0$ then $X(x, y)$ is on a corner.

Notice that if $w_{ij} = 1$ then this is just the gradient covariance of the image and the Harris detector is essentially a local Principle Components Analysis (PCA). In fact Harris doesn't actually compute the eigenvalues but instead a related quantity called the "strength":

$$S = \lambda_1 \lambda_2 - \kappa (\lambda_1 + \lambda_2)^2 \quad (7)$$

$$= \det(M) - \kappa \text{trace}^2(M) \quad (8)$$

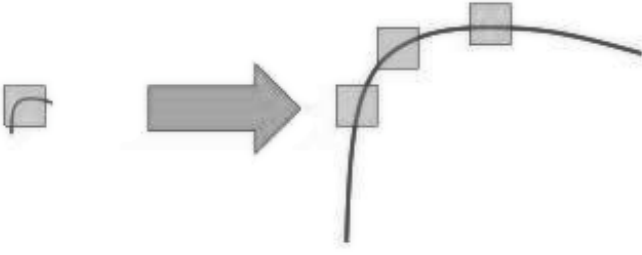


Fig. 3: Harris Detector failing to recognize the right image as a corner.

Hence Bentoutou *et al.* first compute a gradient map of the image using a first order Gaussian derivative filter. They then threshold¹ the gradient map at the average gradient value, thereby extracting only sufficiently "interesting" regions, and compute the strength S for all pixels. They also apply Non-maximum Suppression² (NMS) using a 3×3 window and further threshold the remaining non-zero strength values at a threshold of 1% of maximum observed strength. Finally only the "strongest" n corners are kept.

1.1.1.2 SIFT

One issue with Harris detectors is that they're not invariant to scale (see figure 3). Zahra *et al.*[4] resolves this issue by using the Scale Invariant Feature Transform[5] (SIFT) to identify CPs that, as the name implies, are invariant across multiple scales. SIFT identifies scale invariant and noise robust features of an image, called *keypoints*, by first finding candidate points with high local curvature at multiple scales and then culling according to some heuristics.

It then "describes" these keypoints by a rotation invariant and noise robust representation. The algorithm consists of five steps:

- 1) Scale-space pyramid construction: a sequence of increasingly sub-sampled and more strongly Gaussian filtered images is computed. The sequence of differences of these images is also computed; the sequence of differenced images approximates the multi-scale Laplacian of Gaussians³ (LoG) of the image (see figure 1).
- 2) Keypoint detection: candidate keypoints are points on edges with curvature, i.e. extrema along scale and space dimensions in the LoG pyramid (see figure 1).
- 3) Keypoint selection: candidate keypoints are more precisely localized using an iterative process. Keypoints of low-contrast (therefore sensitive to noise) or on edges

of low curvature⁴ are culled.

- 4) Keypoint orientation assignment: orientation is assigned to each keypoint by taking a weighted majority vote of all gradient orientations in a neighborhood of the keypoint (see figure 4b). Large minority votes (80% of majority) are used to create more keypoints at the same pixel point.
- 5) Keypoint descriptor computation: for each keypoint the descriptor is computed by partitioning the keypoint's neighborhood into 2^k sub-neighborhoods, computing an 8-bin histogram of oriented gradients⁵ (HOG) in each sub-neighborhood, and concatenating (see figure 4d). In Lowe *et al.*[5] $2^4 = 16$ sub-neighborhoods are used to produce an $8 \times 16 = 128$ entry length descriptor. The descriptor is also normalized to unit length in order to make it invariant to luminance (intensity).

SIFT is indeed effective as a CP detector but unfortunately it is patented. Alternatives include Binary Robust Invariant Scalable Keypoints[6], and Oriented FAST and rotated BRIEF[7] (which itself consists of applying Features from accelerated segment test[8] to detect points of interest and Binary Robust Independent Elementary Features[9] to compute descriptors).

1.1.2 Feature Matching

After robust features are identified in the reference image and the displaced images, they need to be matched. For example for SIFT, where the descriptors are designed to be invariant across images, Euclidean distance using a k -d tree⁶ can be used to efficiently match keypoint descriptors. Although this often leads to false-positive matches (Zahra *et al.* resolve this by using Random Sample Consensus (RANSAC)⁷) it's a natural feature matching method. In other cases the matching mechanism is not so straightforward; for a class of algorithms called area-based or intensity-based algorithms, that in fact combine the feature detection and matching step into one, matching involves comparing summaries of patches in the reference image and the displaced image.

1.1.2.1 Normalized Cross-correlation

The simplest strategy for matching by region is to grid-search⁸ the space of possible translational shifts $\Delta x, \Delta y$ and assess the quality of the registration for a given shift using a similarity metric. One such similarity metric is Normalized

⁴The ratio of the eigenvalues of the spatial Hessian of D (i.e. only along dimensions x, y). In fact a quantity not unlike eqn. (8) is computed in order to save having to explicitly find eigenvalues.

⁵Simply the histogram of gradient orientations (polar angle) in a pixel window.

⁶ k -d trees[10], short for k -dimensional trees, are data structures that partition space efficiently in order that searching the tree, insertion into the tree, and deletion from the tree are all, on average, $O(\log n)$ time operations (where n is the number of nodes in the tree at the time of the operation).

⁷RANSAC is a method used to estimate parameters of a model given outliers. In this case Zahra *et al.* use RANSAC at the transform estimation step to eliminate falsely matched keypoint pairs. RANSAC iterates by repeatedly random sampling the putative matching keypoint pairs and fitting a transform model. At a given iteration the fitted transform model is tested against the unused keypoint pairs and evaluated (according to goodness of fit on a subset called the *consensus set*).

⁸Searching a space by taking regular incremental steps along each dimension (e.g. search $U \subset \mathbb{R}^2$ by checking all $x = \Delta x + x_0, y = \Delta y + y_0$ for $(x_0, y_0) \in U$ and for fixed $\Delta x, \Delta y$).

¹Set everything below a threshold to zero.

²Pick the maximum in a neighborhood and set all other values to zero.

³The Laplacian of an image X is $L(x, y) = \partial_x^2 X + \partial_y^2 X$. Since in practice this approximates a noisy signal (second derivative), smoothing by a Gaussian is a necessary prerequisite. Therefore the Laplace of Gaussians (LoG) filter is the Laplacian composed with the Gaussian:

$$LoG(x, y, \sigma) = -\frac{1}{\pi\sigma^4} \left[1 - \frac{x^2 + y^2}{2\sigma^2} \exp \left\{ -\frac{x^2 + y^2}{2\sigma^2} \right\} \right]$$

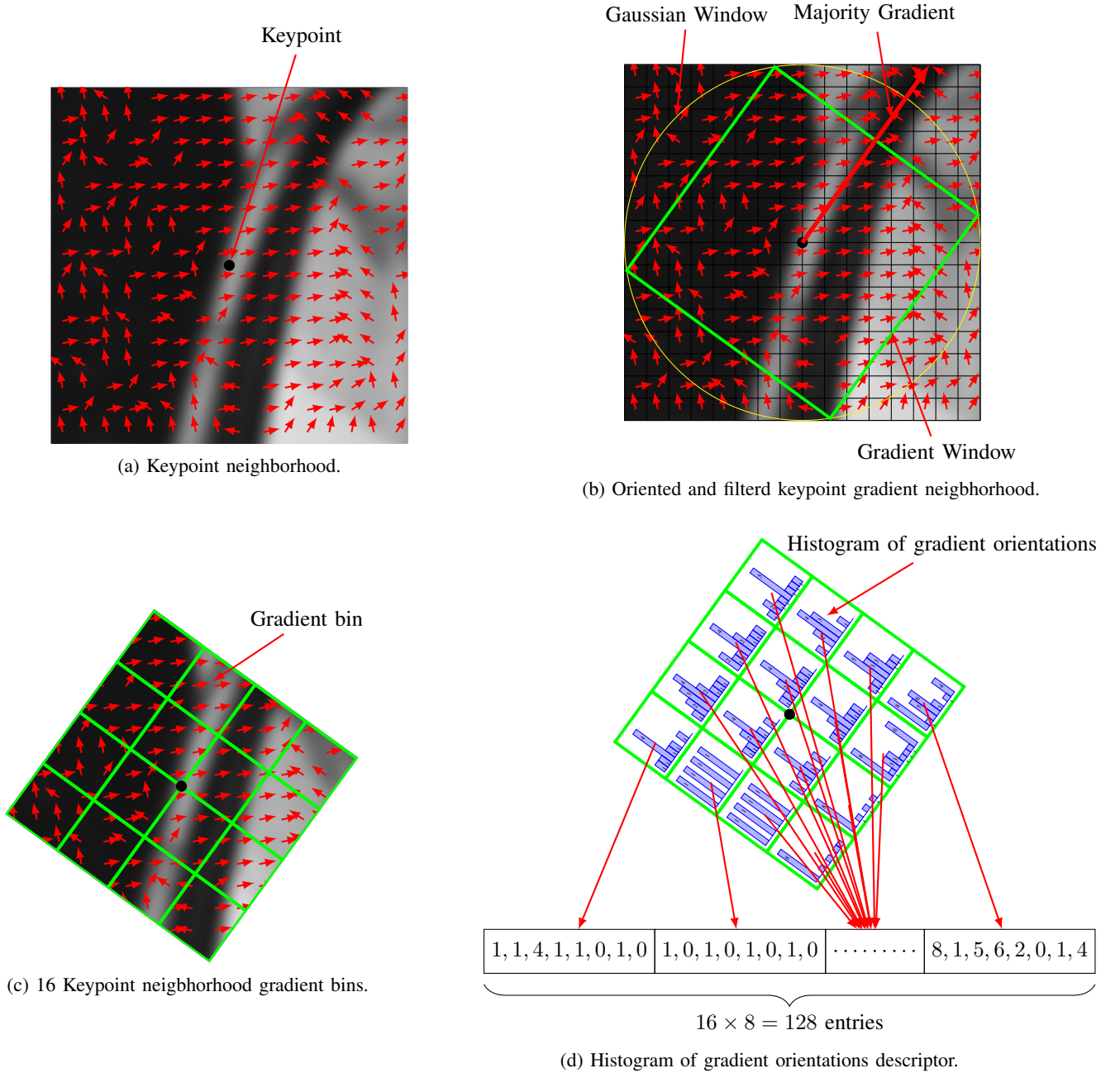


Fig. 4: Sift keypoint descriptor construction.

Cross-correlation (NCC); for two image patches X, Y (with the same width, height) NCC is defined in a straightforward way

$$NCC(X, Y) = \frac{\sum_{x,y} (X(x, y) - \hat{X}) (Y(x, y) - \hat{Y})}{\sqrt{\sum_{x,y} (X(x, y) - \hat{X})^2} \sqrt{\sum_{x,y} (Y(x, y) - \hat{Y})^2}} \quad (9)$$

where \hat{X}, \hat{Y} are mean patch values.

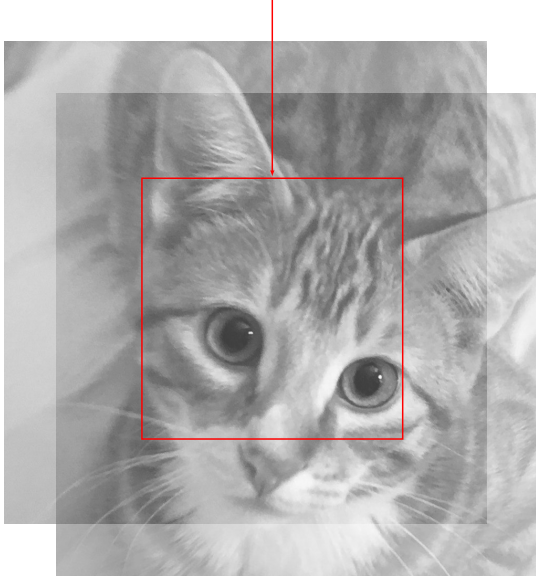
The NCC image registration technique performs a grid-search over possible shifts and compute the cross-correlation of the shifted images (see figure 5a) (usually over a fixed

cross-correlation window rather than the entire image for the sake of computational efficiency). The maximum response as a function $\Delta x, \Delta y$ is the imputed translation between the images (see figure 5b). An alternative but closely related method is *phase correlation*, based on the Fourier shift theorem⁹ (the value of going to Fourier space is availability of highly optimized algorithms for computing the Fourier transform). Let $\mathcal{X}(u, v) = \mathcal{F}\{X(x, y)\}$, $\mathcal{Y}(u, v) = \mathcal{F}\{Y(x, y)\}$ be the

⁹If $\mathcal{F}\{[x_n]\}_k = X_k$ is the k th frequency component of the discrete signal $[x_n]$ then shifting $[x_n]$ by m steps implies

$$\mathcal{F}\{[x_{n-m}]\}_k = X_k \cdot e^{-i2\pi km/N}$$

Fixed cross-correlation window



(a) 100 pixel shifted images.

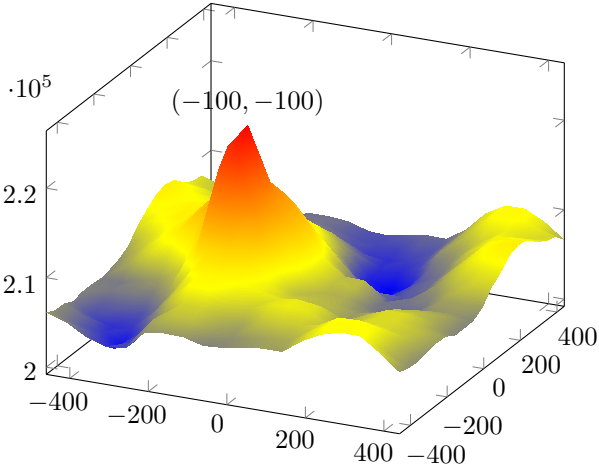
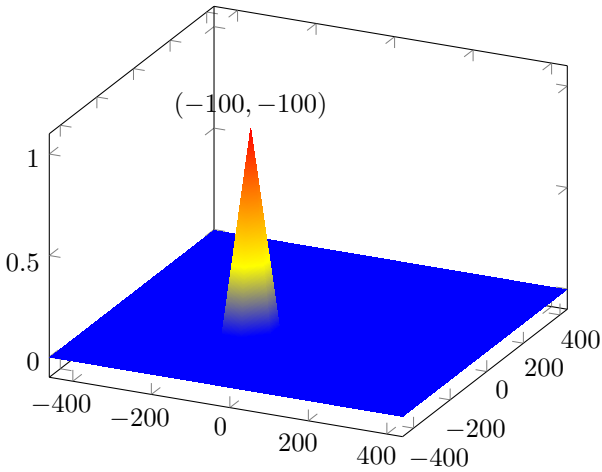
(b) Cross-correlation for various $\Delta x, \Delta y$ shifts, with highest correlation at the shift.(c) Phase-correlation for various $\Delta x, \Delta y$ shifts, with single peak at the correct shift.

Fig. 5: Cross-correlation feature matching.

Fourier transforms of the displaced images. Then

$$\mathcal{X} = \mathcal{Y} e^{-2\pi i (\frac{u\Delta x}{M} + \frac{v\Delta y}{N})}$$

where N, M are the dimensions of the images and the *normalized cross-power spectrum*

$$\begin{aligned} R(u, v) &= \frac{\mathcal{X} \circ \mathcal{Y}^*}{|\mathcal{X}| |\mathcal{Y}^*|} \\ &= \frac{\mathcal{X} \circ \mathcal{X}^* e^{2\pi i (\frac{u\Delta x}{M} + \frac{v\Delta y}{N})}}{|\mathcal{X}| |\mathcal{X}^*| e^{-2\pi i (\frac{u\Delta x}{M} + \frac{v\Delta y}{N})}} \\ &= \frac{\mathcal{X} \circ \mathcal{X}^* e^{2\pi i (\frac{u\Delta x}{M} + \frac{v\Delta y}{N})}}{|\mathcal{X}| |\mathcal{X}^*|} \\ &= e^{2\pi i (\frac{u\Delta x}{M} + \frac{v\Delta y}{N})} \end{aligned} \quad (10)$$

where \circ is Hadamard product¹⁰, $|\mathcal{X}|$ is the magnitude of \mathcal{X} , \mathcal{X}^* is the complex conjugate of \mathcal{X} and we've used the fact that $|e^{iz}| = 1$ for all z . Then the inverse Fourier transform of eqn. (10)

$$\mathcal{F}^{-1} \{R(u, v)\} = r(x, y) = \delta(x + \Delta x, y + \Delta y)$$

is a single peak (see figure 5c) at $(\Delta x, \Delta y)$. The simplest implementation of NCC only identifies translations but it can be extended to affine transforms[11].

1.1.3 Mutual Information

In general NCC fails for noisy images (the to-be-registered image is much noisier than the reference image). Mutual information (MI) methods have been successfully employed in such cases; MI is also robust toward changes in light intensity, pixel color, noise and can fit large transformations. The mutual information of two random variables X_1, X_2 is defined

$$I(X_1, X_2) := H(X_1) + H(X_2) - H(X_1, X_2)$$

where H is Shannon-entropy¹¹. In this context X_1 is reference image and X_2 is the to-be-registered image. The central challenge in computing MI is in estimating joint $H(X_1, X_2)$ and marginal $H(X_1)$ entropies (which are functions of the joint $P(X_1, X_2)$ and marginal $P(X_1)$ probabilities). In practice this is done using either Kernel Density Estimation¹² (KDE) or the joint histogram (i.e. number of common pixel values at same pixel coordinates normalized by total number of pixels). Searching for the registration parameters θ that maximize

$$^{10} \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \circ \begin{bmatrix} b_{11} & b_{12} \\ b_{21} & b_{22} \end{bmatrix} := \begin{bmatrix} a_{11} b_{11} & a_{12} b_{12} \\ a_{21} b_{21} & a_{22} b_{22} \end{bmatrix}$$

¹¹Shannon entropy can be interpreted as the number of bits L necessary to encode a random variable X distributed according to distribution P . It is defined as

$$H(X) := \sum P(x) \log \left[\frac{1}{P(x)} \right]$$

i.e. $P(x_i) = 1/2^L$ where L is the number of bits.

¹²Alternatively known as Parzen windowing, the kernel density estimator \hat{P} of a distribution $X \sim P$ given a set of points $\{x_1, x_2, \dots, x_n\}$ is defined as

$$\hat{P}(x) = \frac{1}{nh} \sum_{i=1}^n K\left(\frac{x - x_i}{h}\right)$$

where K is a kernel (e.g. Gaussian) and h is the *kernel bandwidth*.

$I_\theta := I(X_1, X_2(\theta))$ exhaustively is generally costly (especially if more than just translation parameters are required). Ritter *et al.* **ritter1999** use simulated annealing to efficiently search the space of all possible parameters. Simulated annealing iterates on the current candidate minimum I_θ by "jumping" to potentially new minima in a way that prioritizes near jumps but allows for the possibility of far jumps. They use the Cauchy distribution as the minimum proposal function:

$$G_k(d) = \frac{t_k}{\pi (d^2 + t_k^2)}$$

where t_k is a parameter that over time encourages near jumps more and more strongly (called the temperature), and d is the parameter jump distance. To allow escape from local minima of I_θ it is necessary to occasionally make a jump from the current parameter estimate θ_k to a proposed value θ_{k+1} that is "worse" (i.e. for which $I(\theta_{k+1}) < I(\theta_k)$). This accomplished by querying an *acceptance function* A_k on every such instance:

$$A_k(\theta_k, \theta_{k+1}) = \begin{cases} 1 & \text{if } I(\theta_{k+1}) \geq I(\theta_k) \\ \frac{e^{I(\theta_k) - I(\theta_{k+1})}}{C t_k} & \text{otherwise} \end{cases}$$

1.1.4 Transform Estimation

Furthermore, the transformation to be estimated should incorporate prior knowledge about the motion model but simultaneously lead to a tractable estimation problem (i.e. reasonable number of parameters).

1.2 Gaussian Process

They innovation of this technique is the fully Bayesian approach; they marginalize over the unknown high-resolution image in order to infer the low-resolution image registration parameters. Further, by virtue of marginalization, they're able to estimate the PSF along with the image registration parameters.

REFERENCES

- [1] Y. Bentoutou, N. Taleb, K. Kpalma, and J. Ronsin, "An automatic image registration for applications in remote sensing," *IEEE transactions on geoscience and remote sensing*, vol. 43, no. 9, pp. 2127–2137, 2005 (cit. on p. 1).
- [2] C. G. Harris, M. Stephens, *et al.*, "A combined corner and edge detector.," in *Alvey vision conference*, Citeseer, vol. 15, 1988, pp. 10–5244 (cit. on p. 1).
- [3] H. P. Moravec, "Obstacle avoidance and navigation in the real world by a seeing robot rover.," Stanford Univ CA Dept of Computer Science, Tech. Rep., 1980 (cit. on p. 1).
- [4] Z. Hossein-nejad and M. Nasri, "Image registration based on sift features and adaptive ransac transform," in *2016 International Conference on Communication and Signal Processing (ICCSPP)*, Apr. 2016, pp. 1087–1091. DOI: [10.1109/ICCSPP.2016.7754318](https://doi.org/10.1109/ICCSPP.2016.7754318) (cit. on p. 3).
- [5] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International journal of computer vision*, vol. 60, no. 2, pp. 91–110, 2004 (cit. on p. 3).
- [6] S. Leutenegger, M. Chli, and R. Siegwart, "Brisk: Binary robust invariant scalable keypoints," in *2011 IEEE international conference on computer vision (ICCV)*, Ieee, 2011, pp. 2548–2555 (cit. on p. 3).
- [7] E. Rublee, V. Rabaud, K. Konolige, and G. R. Bradski, "Orb: An efficient alternative to sift or surf.," in *ICCV*, Citeseer, vol. 11, 2011, p. 2 (cit. on p. 3).
- [8] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European conference on computer vision*, Springer, 2006, pp. 430–443 (cit. on p. 3).
- [9] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: Binary robust independent elementary features," in *European conference on computer vision*, Springer, 2010, pp. 778–792 (cit. on p. 3).
- [10] J. L. Bentley, "Multidimensional binary search trees used for associative searching," *Commun. ACM*, vol. 18, no. 9, pp. 509–517, Sep. 1975, ISSN: 0001-0782. DOI: [10.1145/361002.361007](https://doi.org/10.1145/361002.361007). [Online]. Available: <http://doi.acm.org/10.1145/361002.361007> (cit. on p. 3).
- [11] R. Berthilsson, "Affine correlation," in *Proceedings. Fourteenth International Conference on Pattern Recognition (Cat. No.98EX170)*, vol. 2, Aug. 1998, 1458–1460 vol.2. DOI: [10.1109/ICPR.1998.711979](https://doi.org/10.1109/ICPR.1998.711979) (cit. on p. 5).