Evaluation of Half Gaussian Filter Rotation for Edge Detection

Baptiste Magnier, Behrang Moradi, Philippe Carré

EVALUATION OF HALF GAUSSIAN FILTER ROTATION FOR EDGE DETECTION

Baptiste Magnier[1]*, Behrang Moradi*[1] *and Philippe Carré*[2]

[1] IMT Mines Alès, LGI2P, 6. av. de Clavières 30100 Alès, France
[2] XLIM, UMR CNRS 7252, Univ. de Poitiers, 11 Bd. Marie et Pierre Curie, 86073 Poitiers, France

ABSTRACT

In image processing, contours can be extracted by linear convolution of the image with first-order derivative filters. The oriented half Gaussian filters are directed in all desired directions to analyze the images. Thus, they are useful for detecting contours or extracting precise orientations. The process of filter orientation requires the application of a rotation technique in a two-dimensional discrete domain of different scientific strategies: interpolation (which can modify image information) or discrete geometry (without modification of the image values). In this paper, different methods of discrete rotating the half Gaussian filter are compared and evaluated to detect contours in synthetic and real images. Results are qualitatively and quantitatively compared, validating which rotation technique is most beneficial for edge detection.

Index Terms— Oriented filters, rotation, interpolations.

1. INTRODUCTION AND MOTIVATION

Easy to use, Gaussian half-filters are reliable for image analysis. These narrow filters, directed in all desired directions around each pixel, are useful for detecting the contours or extracting precisely the orientations, even for very noisy images. The oriented half-filters [1, 2] were inspired by the well-known and popular *steerable filters* [3] using full 2D Gaussian (isotropic) where the calculation of the norm of the gradient corresponds at the energy in the direction of the maximum response of the filter. Thus, Freeman and Adelson have shown that the first derivative of the directional Gaussian $\mathcal{G}_{\sigma,\theta}$ of θ can be generated by a linear combination of the direct rotation of the derivatives of the basic isotropic Gaussian with respect to x and y axis:

$$\mathcal{G}_{\sigma,\theta}(x,y) = \cos(\theta) \cdot \frac{\partial \mathcal{G}_\sigma}{\partial x}(x,y) + \sin(\theta) \cdot \frac{\partial \mathcal{G}_\sigma}{\partial y}(x,y), \qquad (1)$$

where (x,y) represent the coordinates of the pixel and σ is the standard deviation of the Gaussian. Half-filters allow to estimate contour information in several directions up to 360°, unlike full Gaussians that are symmetrical and only get information up to 180° [4].

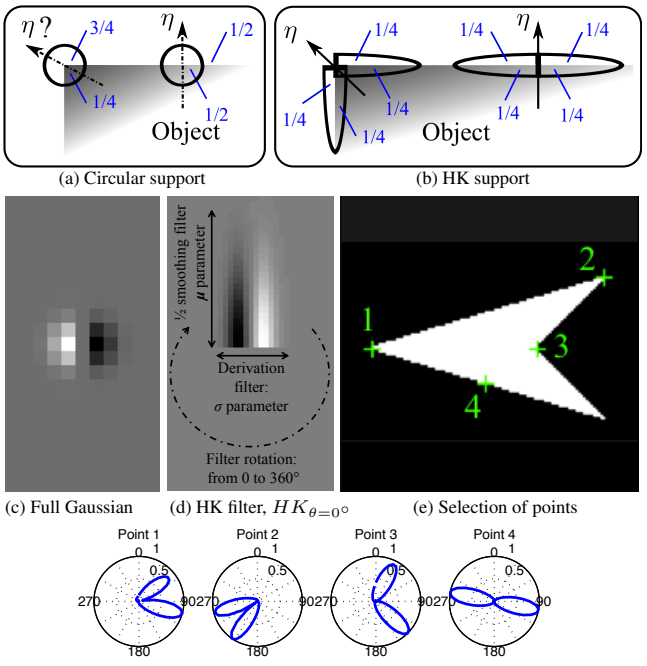
Several contours can cross a single pixel, for example a pixel positioned at a corner where there are several directions, as illustrated in Fig. 1 (e) points 1 to 3. These directions can be estimated using half-filters; they are useful and effective for restoring images via PDE [4], corner detection [5] or descriptors [6]. As a result, the filter must be narrow to maintain the most robust precision possible. However, since the filter is discrete and in 2D, the rotation technique of the filter can alter the quality of the filter, which could influence the detected contours; especially when the filter has a support of a few pixels. In this work, we propose to study different families of rotations and to examine how they influence, impact or degrade the filter when it is oriented. Therefore, different rotation methods are described in the following sections and evaluated according to the quality of the contours obtained.

2. DERIVATIVE HALF GAUSSIAN KERNELS

Gaussian kernels are regularly used for their effectiveness in edge detection [7]. Nevertheless, weaknesses can be observed in the presence of blur or noise and at the level of the corners and small objects in the image. Edge detection techniques using elongated Gaussians are effective for correctly detecting large linear structures [8], but the small elements in the image are considered as noise and their contours are not extracted. Consequently, the accuracy of the detected contour points greatly decreases at the corner levels and the non-rectilinear object portions [2]. To bypass this undesirable effect, an anisotropic edge detection method was developed in [1] and analyzed in more detail in [2]. Indeed, the proposed technique is able to detect corners and intersecting contours due to two elongated filters and directed in two different directions. The main idea is to *cut* the anisotropic Gaussian kernel using a Heaviside function and oriented this filter in all directions around the considered pixel, from 0 to 360°. A half kernel (HK) can be built by combining a semi Gaussian (i.e. a truncated Gaussian) on the one hand and its first derivative on the other hand. They are defined as follows:

- a semi/truncated Gaussian for the smoothing:
  $\mathcal{G}(s) = H(s) \cdot \mathbf{e}^{\frac{s^2}{2\cdot\mu^2}}$, with $\mu \in \mathbb{R}_+^*$ and $s \in \mathbb{R}$,
  where H represents the Heaviside function,
- a first derivative of a Gaussian (derivative of $\mathcal{G}$) :
  $\mathcal{G}_1(s) = s \cdot \mathbf{e}^{\frac{s^2}{2\cdot\sigma^2}}$, with $\sigma \in \mathbb{R}_+^*$ and $s \in \mathbb{R}$.

For signal and image processing, s represents an integer. Fig. 1(d) shows an example of HK, constructed with the two functions, respectively, $\mathcal{G}_1$ at the horizontal and $\mathcal{G}$ at the vertical. In order to create an anisotropic (elongated) filter and smooth majoritarily in the edge direction to robustly capture



(a) Circular support (b) HK support

(c) Full Gaussian (d) HK filter, $HK_{\theta=0^\circ}$ (e) Selection of points

(f) Energy computed with the the HK_θ in all directions for selected points in (e)

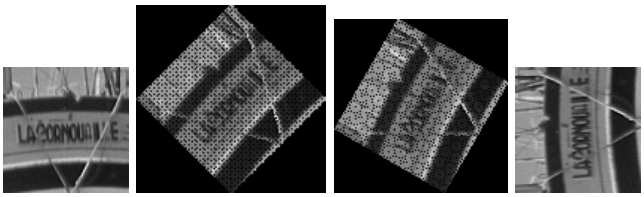
Fig. 1. Representation of filter supports concerning edges and corners and selection of points before applying the HK_θ.

edges [4, 2], the support of the smoothing half-filter must be greater than the support of the filter containing the derivative, that is to say $\mu > \sigma$. Then, to obtain a rotated version of the filter called HK_θ, the HK filter is directed in several directions θ from 0 to 360°. The convolution of the image I with HK_θ (i.e. $I * HK_\theta$) allows to compute a derivative information at each desired direction (as shown in Fig. 1 (e)-(f)).

In order to better understand this edge extraction technique [2], the HK filter support at a straight contour is equivalent to 1/2 on both sides of the edge, as shown in Fig. 1 (b). On the contrary, for a corner point with an angle of 90°, the support of the half-filter remains 1/2 on both sides of the edge, whereas it is around 1/4 and 3/4 concerning complete filters (e.g. isotropic Gaussian, see Fig. 1 (a)). Finally, HK corresponds to the derivation of an oriented filter; consequently, its responses are either positive or negative. Therefore, for each pixel of coordinates (x, y), the gradient $|\nabla I|$ is the maximum value minus the minimum value of $I * HK_\theta$ among all the directions θ:

$$\begin{cases} |\nabla I|(x,y) & = \max\limits_{\theta\in[0,360^\circ[} I * HK_\theta(x,y) - \min\limits_{\theta\in[0,360^\circ[} I * HK_\theta(x,y) \\ \theta_1(x,y) & = \operatorname{argmax}\limits_{\theta\in[0,360^\circ[} (I * HK_\theta(x,y)) \\ \theta_2(x,y) & = \operatorname{argmin}\limits_{\theta\in[0,360^\circ[} (I * HK_\theta(x,y)) \\ \eta(x,y) & = \frac{\theta_1+\theta_2}{2} \end{cases} \tag{2}$$

Thus, for each pixel, θ_1 and θ_2 are computed, they correspond to the directions of the contours, tied to maximum and the minimum value of $I * HK_\theta$ among all the directions θ for each pixel, as in Fig. 5 for two edge directions at 0° and 185° respectively. These directions are useful and efficient



(a) Original (b) 45° (c) 60° (d) 90°

Fig. 2. Image rotation directly using eq. 3.

for image restoration via PDE [4], corner detection [5] or image descriptor [6]. Finally, to obtain thin edges, as in [7], the local maxima suppression operation is processed in the η orientation, corresponding to the bisector between θ_1 and θ_2.

The edge detection technique using oriented truncated filters requires to rotate the HK filter in two dimensions[1]. Nevertheless, the two-dimensional filter is considered as an image with positive and negative values before applying the convolution of itself with the image (details in [2]). The objective of this work is to estimate which rotation method of the filter is the most reliable to extract contours. The following section presents different rotation methods used in image processing which will be evaluated within the context of edge detection.

3. IMAGE ROTATION METHODS

3.1. Rotation with image shear

Geometric transformations of a signal or an image are common operations. Nevertheless, these operations, which are correctly defined in the continuous domain, do not become straightforward in the discrete domain such as a digital image. Therefore, the rotation of the images considers a discrete 2D signal (i.e., the image) to provide a second discrete 2D signal. However, the relationship between the pixels of these two images is not explicit.

The conventional rotation transformation of angle θ is described by the following matrix:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}, \tag{3}$$

where (x, y) represent the coordinates of a pixel of the input image and (x', y') correspond to the coordinates of the pixel at its new position after rotation.

A rotation transformation is easily implemented using a matrix multiplication. However, this requires many arithmetic operations with floating numbers and calculations using trigonometric functions. After these operations, a certain number of pixels of the new image are unanswered; these points correspond to empty pixels or holes (as shown in Fig. 2, except for rotations concerning angles modulo 90°). Therefore, an interpolation process is also applied. In order

[1] In [1], the rotation of an angle θ is applied to the image, then the rotated image is filter using $HK_{\theta=0^\circ}$ and is rotated in reverse with an angle of $-\theta$; corresponding to two rotations. The study carried by this work concerns the rotation of the filter, after the image is filtered with the oriented filter HK_θ.

to overcome these transformations, which are complicated to implement and can modify image information, the rotation matrix can be decomposed into 3 steps. Indeed, the rotation described in eq. 3 is rewritten as the product of 3 matrices [9] where each matrix corresponds to a shear transformation:

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} 1 & -\tan(\frac{\theta}{2}) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ \sin(\theta) & 1 \end{pmatrix} \begin{pmatrix} 1 & -\tan(\frac{\theta}{2}) \\ 0 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix}. \quad (4)$$

Each shear corresponds to a 1D signal translation, in total, two horizontal shears and one vertical, as illustrated in Fig. 3. The major benefit of this technique is to preserve high spatial frequencies. Even though the three-shear rotation decomposition provides less complex algorithms than the eq. 3, interpolations or pixel placements are required at each shear step. Thus, there are two important techniques for rotating images: interpolations [10, 11, 12] and discrete reversible rotations without loss of pixels [13, 14].

3.2. Rotations using interpolations

Each shear step requires an interpolation model for resampling [10, 11, 12]. There are several interpolation techniques, three of them are described here, one copies existing pixels (Nearest), another uses a local mean (Bilinear) while the last one averages and enhances the contours in the same way (Bicubic) using both a smoothing and a derivation.

3.2.1. Nearest Neighbor

Each pixel value in the resampled image is determined by simply copying the value of the nearest input pixel. This method preserves the original pixel values. The approximation is discontinuous and has no regularity; its mathematical expression is given by:

$$f_{\mathcal{N}}(x) = \begin{cases} 1, & \text{si } |x| < 1 \\ 0, & \text{elsewhere.} \end{cases} \quad (5)$$

Since this algorithm is the only one that does not insert new colors in the result, it produces very visible block or stair effects on the result during the rotation process.

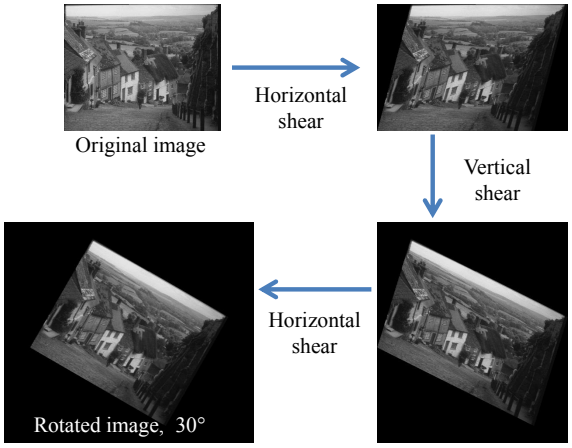


Fig. 3. Successive shears and bilinear interpolations.

3.2.2. Bilinear

Often used, this sampling is computed by the linear-weighted average of the nearest pixel values of the pixel considered [10]. Indeed, in 1D, this technique can be implemented by the following triangle function $f_{\mathcal{L}}$:

$$f_{\mathcal{L}}(x) = \begin{cases} 1 - |x|, & \text{si } |x| < 1 \\ 0, & \text{elsewhere.} \end{cases} \quad (6)$$

The bilinear interpolation is easy to compute and minimizes aliasing effects, but introduces significant blur in the image.

3.2.3. Bicubic

This function is composed of 3rd degree piece polynomials [15]. The weighting factors for the average of the input pixels are calculated using a cubic function of the pixel distance:

$$f_{\mathcal{B}ic}(x) = \begin{cases} (a+2)|x|^3 - (a+3)|x|^2 + 1, & \text{if } 0 \leq |x| < 1 \\ a|x|^3 - 5a|x|^2 + 8a|x| - 4a, & \text{if } 1 \leq |x| < 2 \\ 0, & \text{elsewhere,} \end{cases} \quad (7)$$

where $a = -0.5$ is a parameter that produces 3rd order convergence [15]. Bicubic resampling is one of the most used in image resampling. Indeed, as this interpolation method contains negative coefficients, it makes it possible to reduce both the aliasing effect and the blur in the result image compared to the method previously detailed.

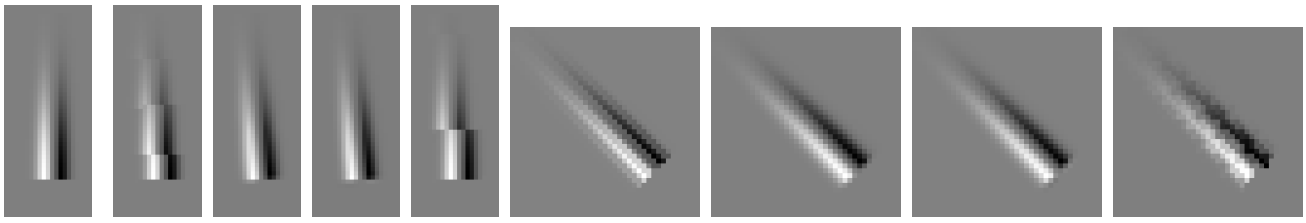
There exist other interpolation methods for resampling images [10, 11, 12, 16]; we were interested in these last three for their property of copy of pixels (Nearest), averaging (bilinear) and enhancement of contours (bicubic) when the half Gaussian filter must be rotated (Fig. 4).

3.3. Discrete rotation

Since previous transformations behave as low-pass filters, important image structures can be lost/smoothed by using them, especially high-frequency information. Instead of interpolating, the basic idea of discrete rotations is to replace each horizontal and vertical shear transformation with one to one pixel displacements [13, 14]. The called quasi-shear transformation $\mathcal{Q}$ defined by:

$$\mathcal{Q}(x, y)) = \begin{pmatrix} x + \lfloor \frac{a \cdot y + c}{b} \rfloor \\ y \end{pmatrix}, \quad (8)$$

$\frac{a}{b}$ represents the slope of quasi-shear, c is the translation factor, with a, b, and c integers, such that $c>0$, where $\lfloor \cdot \rfloor$ defines the function that returns the integer part of a number ($floor$ function). The function $\mathcal{Q}$ is applied for a horizontal transformation; during a vertical transformation, it suffices to invert the coordinates x and y and thus return to the eq. 4. The advantage of a discrete rotation by translation is that it simply moves the pixels without modifying their values (illustrated in Fig. 4(e)), and especially, that the inverse rotation corresponds exactly to the original image (i.e., bijective transformation). The objective of the following study is to evaluate which rotation technique applied to the half-filters is the most reliable to detect contours. Note that we proposed to use



(a) $\theta = 0°$ Original (b) $\theta = 5°$ Nearest (c) $\theta = 5°$ Bilinear (d) $\theta = 5°$ Bicubic (e) $\theta = 5°$ Discrete (f) $\theta = 45°$ Nearest (g) $\theta = 45°$ Bilinear (h) $\theta = 45°$ Bicubic (i) $\theta = 45°$ Discrete

Fig. 4. Half Gaussian kernels rotated at 5° and 45° with the following parameters: $\sigma = 1.53$ et $\mu = 13.11$.

discrete rotation as part of the definition of bases of oriented wavelet packets [17].

Visually, the filters are shown in Fig. 4 for rotations at 5° and 45°. The bilinear and bicubic rotations allow to obtain a smoothed rotated filter, whereas the copied pixels are clearly visible with the nearest rotation. Finally, the discrete rotation brightly marks a shear of the 5° filter (filter cut in the middle) and the displaced pixels are noticeably apparent at 45°.

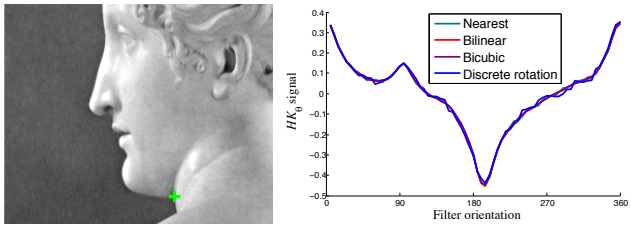
4. EVALUATION

Firstly, the Fig. 4 illustrates the difference between the different rotation techniques of the filter, and the Fig. 5 shows the signal obtained at the level of one pixel using theses rotation methods. Thus, the curves using the bilinear and bicubic formulas are better smoothed than with the nearest and the discrete techniques, which create irregular signals.

Various supervised methods have been proposed in the literature to evaluate edge detectors [18] and HK has been compared to other simple, oriented and anisotropic edge detectors in [18], [19] and [2]. Hereafter, let G_t be the reference contour map corresponding to the ground truth and D_c the detected contour map of an image I. Comparing pixel by pixel G_t and D_c, a basic evaluation is composed of statistics:

- True Positive (TP), commun points of G_t and D_c,
- False Positive (FP), spurious detected edges of D_c,
- False Negative (FN), missing boundary points of D_c,
- True Negative (TN), common non-edge points.

It is clearly proved that poorly located or missing pixels should be penalized according to the distance from the position where they should be localized. Also, as demonstrated in [18], the evaluation of FP and FN should not be symmetrical, because such a penalty could alter the visibility of the



(a) Image 261×213 (b) Obtains signals, σ=1.53, μ=10.96

Fig. 5. Comparison of signals obtained with different rotation techniques for the same pixel in Paulina image (green cross).

outlines of the desired objects in an objective evaluation (see [18]): some measures calculate a large error for a single FP at a sufficiently large distance, while many desired contours are missing, but unfortunately, they are not penalized enough. Thus, described in [19], the normalized $\mathcal{N}$ edge detection evaluation measure is, for $FN > 0$ or $FP > 0$:

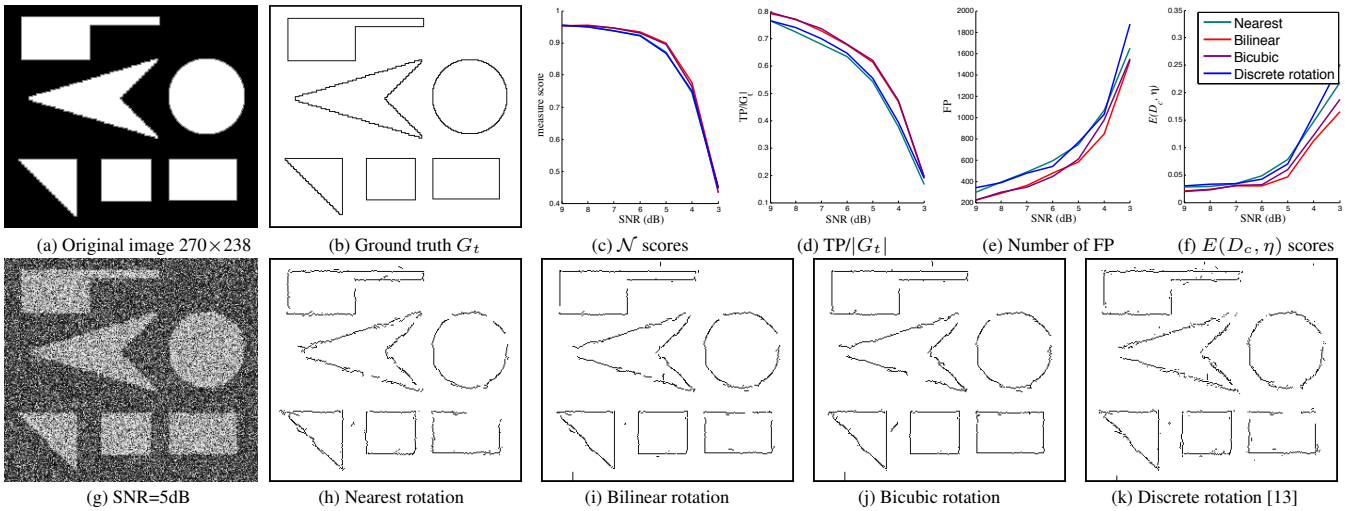
$$\mathcal{N}(G_t, D_c) = \frac{1}{FP + FN} \cdot \left[\frac{FP}{|D_c|} \cdot \sum_{p \in D_c} \frac{1}{1 + \kappa_{FP} \cdot d^2_{G_t}(p)} + \frac{FN}{|G_t|} \cdot \sum_{p \in G_t} \frac{1}{1 + \kappa_{FN} \cdot d^2_{D_c}(p)} \right], \quad (9)$$

where $(\kappa_{FP}, \kappa_{FN}) \in]0, 1]^2$ represent two scale parameters [19], $|\cdot|$ denotes the cardinality of a set, and $d_A(p)$ is the minimal Euclidian distance between a pixel p and a set A [20]. Therefore, the measure $\mathcal{N}$ calculates a standardized dissimilarity score; the closer the evaluation score is to 1, the more the segmentation is qualified as suitable. On the contrary, a score close to 0 corresponds to a poor detection of contours, finally, if FP=FN=0, then $\mathcal{N}$ =1.

The objective here is to get the best contour map in a supervised way. The ground truth images are available in Figs. 6(b) and 7(f). The 4 rotation methods described above are applied to the filters in order to compare the obtained segmentation. For that, the contours are extracted after a suppression of the local non-maxima (cf eq. 2), then a threshold by hysteresis is applied to obtain a binary segmentation [7]. Theoretically, to be objectively compared, the ideal contour map of a measure must be a D_c at which the supervised evaluation gets the highest score [18, 19]. For this, a double loop for the hysteresis thresholding is applied in order to obtain the best segmentation as a function of $\mathcal{N}$ (i.e., the best score of $\mathcal{N}$ for a given contour detection). For each better segmentation tied to $\mathcal{N}$, the false positives (FP) and percentage of true positives relative to the total number of edge pixels of G_t are also displayed (TP/G_t). Also, the last evaluation measure concerns the gradient angle, η [2]. Once D_c is created, considering $\mathcal{C}_{D_c}$, the set of contour chains in D_c (i.e., at least 2 pixels per chain), the gradient evaluation is computed as follows:

$$E(\mathcal{C}_{D_c}, \eta) = \frac{1}{|\mathcal{C}_{D_c}|} \cdot \sum_{p \in \mathcal{C}_{D_c}} \sum_{d_k \in W} \left[1 - \frac{|90° - |\overrightarrow{\eta_p} - \overrightarrow{\eta_{d_k}}| \,|}{90°} \right] / c_k,$$

where d_k represents a contour pixel belonging to W, a 3×3 window centered on p, $\overrightarrow{\eta_{d_k}}$ the gradient orientation of d_k and c_k the number of contour pixels in W, minus the central pixel. This evaluation linearly ranges from 0 for identical angles of $\overrightarrow{\eta_p}$ and $\overrightarrow{\eta_{d_k}}$ to 1 for angles that differs. Note that $\overrightarrow{\eta_{d_k}}$ and $\overrightarrow{\eta_p}$



(a) Original image 270×238 (b) Ground truth G_t (c) $\mathcal{N}$ scores (d) TP/$|G_t|$ (e) Number of FP (f) $E(D_c, \eta)$ scores

(g) SNR=5dB (h) Nearest rotation (i) Bilinear rotation (j) Bicubic rotation (k) Discrete rotation [13]

Fig. 6. Comparison and evaluation of edge detection using diffrent rotation techniques for the oriented half-Gaussian kernel. Filter parameters are: $\sigma = 0.7$ and $\mu = 8.81$, the angular step is of 5°. Detected edges are tied to the noisy image of SNR=5dB.

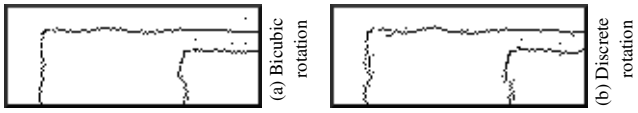
angles belong to $[0; 180°[$ and when one direction approximates 0 and the other direction 180°, the evaluation of these two directions remains close to 0. These scores are presented in Figs. 6 (d-f) and 7 (b-e) according to the *SNR* of the image, ranging from 9dB for the less noisy up to 3dB for the most deteriorated. Therefore, the scores of $\mathcal{N}$ should decrease monotonously with respect to the level of degradation (white Gaussian noise); the filter rotation technique with the highest scores is considered the most reliable.

4.1. Synthetic images

The synthetic image presented in Fig. 6(a) is composed of geometric shapes. They contain step edges, however, an intermediate value gray pixel has been added at each contour. These gray pixels represent an accurate G_t of the original, visible in Fig. 6(b). Regarding the scores of $\mathcal{N}$ displayed in Fig. 6(d), the curves tied to the nearest and discrete rotation overlap. Scores of these two rotation techniques are lower than the bilinear and bicubic methods applied to this synthetic image. On the contrary, the bilinear and bicubic methods obtain similar scores for the evaluation function $\mathcal{N}$, regardless of the noise level, just like the number of TPs; however, the bilinear rotation obtains less FPs. Visually, the contours presented are those obtained in the image presented in Fig. 6(b), with SNR=5dB. In this experiment, the used filter is thin, with few pixels for the derivation support ($\sigma = 0.7$ and $\mu = 8.81$), the contours extracted using the nearest and discrete rotation are less continuous than the other two techniques. Regarding the nearest rotation, an important contour is missing on the rectangle at the bottom right. The discrete rotation creates many FPs, this is explained by the displacement of pixels during the rotation process of the filter, since this mixes the coefficients of the filter (see Fig. 4(i)). On the other hand, the gradient direction evaluation $E(\mathcal{C}_{D_c}, \eta)$ indicates that the bilinear and bicubic methods obtain more regular/smoothed gradient directions along contours chains.

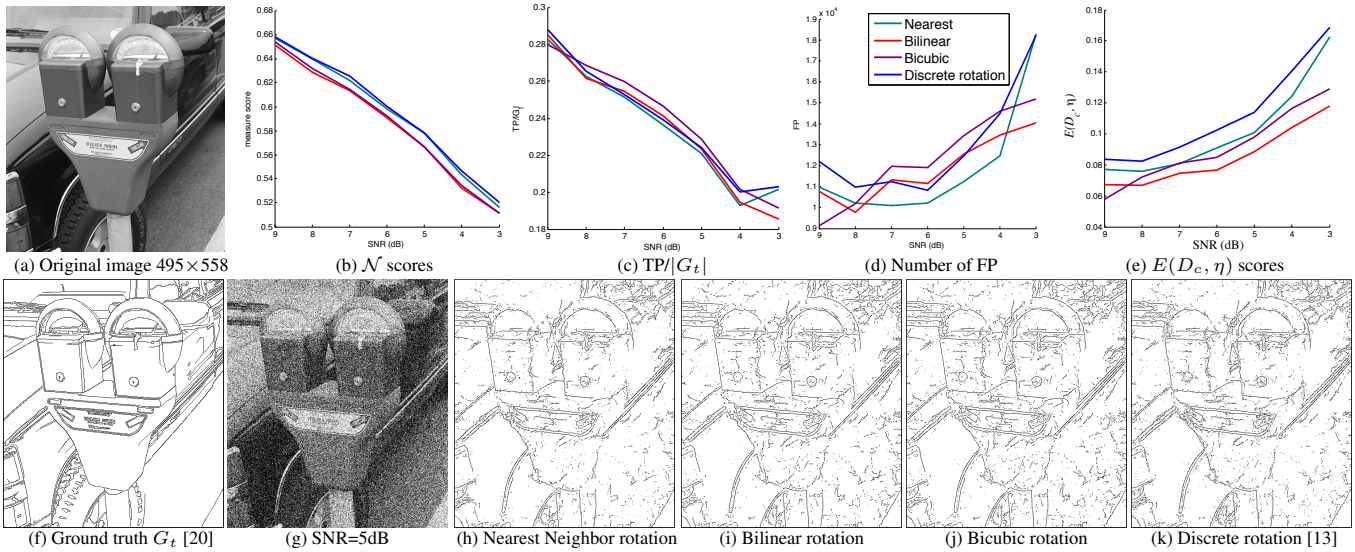
4.2. Real images

The image presented in Fig. 7(a) is tied to its ground truth G_t in Fig. 7(f). Reference contours were created using filters of type [-1 0 1], then missing contours were added manually, and many unwanted contours were also deleted (detailed in [20]). In contrast to the synthetic image, the scores of $\mathcal{N}$ displayed in Fig. 7(b) are to the advantage of the nearest and the discrete rotation techniques, but not for the number of FP and TP and concerning $E(\mathcal{C}_{D_c}, \eta)$. Indeed, the bilinear and bicubic rotation method obtains scores of $\mathcal{N}$ lower than the nearest and the discrete rotation. Visually, the presented contours are those obtained in the Fig. 7(b), with SNR=5dB. The obtained segmentation with the discrete rotation of the filter is less disturbed by the FP than the other methods, the objects are better separated than using the other rotation techniques. However, it is notable that FPs are present on either side of correctly detected contours; this is due to the rotation of the filter for pixels close to the contours. As the filter coefficients are mixed, they can detect false contours according to their new positions in the rotated filter. In terms of precision (TP/$|G_t|$), the bicubic technique is reliable, but creates too much FPs, contrary to the nearest which does not obtain enough TP points, but performs for with less of FPs. On the contrary, the bilinear method obtains better gradient orientation. Finally, the edge detection presented in Fig. 8 illustrates that, even for a larger filter, the extracted contours are less continuous with the discrete technique than with the bilinear (original image in Fig. 6(a), SNR = 4dB) . This example illustrates that the rotation method does not depend to the filter parameters.



(a) Bicubic rotation (b) Discrete rotation

Fig. 8. Crop of edge detection, σ=4 and μ=13, SNR=4dB.



(a) Original image 495×558 (b) $\mathcal{N}$ scores (c) TP/$|G_t|$ (d) Number of FP (e) $E(D_c, \eta)$ scores

(f) Ground truth G_t [20] (g) SNR=5dB (h) Nearest Neighbor rotation (i) Bilinear rotation (j) Bicubic rotation (k) Discrete rotation [13]

Fig. 7. Comparison and evaluation on a real image. Filter parameters: $\sigma = 1.53$ and $\mu = 10.96$ and the angular step is of 5°.

5. CONCLUSION

This paper compares different rotation techniques of half Gaussian filters before convoluting them to images. In this context, the nearest, bilinear, bicubic and discrete filter rotation methods were evaluated. Conventionally, these rotation methods are used to obtain a rotated image with the same quality of visual rendering; this study illustrates the interest of choosing the appropriate rotation method for an oriented semi-filter used in edge detection. The bilinear and bicubic methods act as a low-pass filter, interpolating pixels; on the other hand, the discrete rotation only moves the pixels, creating no new value. Experimental results show that bicubic and bilinear rotations are more accurate for detecting step edges and, usually, they obtain more regular/smoothed gradient directions along contours chains, while the discrete rotation method is more efficient when the image has ramp edges. Without interpolation, the latter preserves the same number and the same coefficient values of the filter, regardless of the rotation applied, avoiding, especially, a low pass effect. A deeper investigation should be carried out by studying which interpolation technique (e.g. Bicubic, or Bilinear) should be adapted as a function of the edge type (step, ramp, ridge...).

6. REFERENCES

[1] P. Montesinos and B. Magnier, "A new perceptual edge detector in color images," in *Advanced Concepts for Intelligent Vision Systems*. Springer, 2010, pp. 209–220.

[2] B. Magnier, "An objective evaluation of edge detection methods based on oriented half kernels," *ICISP*, pp. 80–89, 2018.

[3] W. T. Freeman and E. H. Adelson, "The design and use of steerable filters," *IEEE TPAMI*, vol. 13, pp. 891–906, 1991.

[4] B. Magnier and P. Montesinos, "Evolution of image regularization with pdes toward a new anisotropic smoothing based on half kernels," in *Image Processing: Algorithms and Systems XI*. Int. Soc. for Optics and Photonics, 2013, vol. 8655, p. 86550M.

[5] H. Abdulrahman, B. Magnier, and P. Montesinos, "Oriented asymmetric kernels for corner detection," in *EUSIPCO*, 2017, pp. 778–782.

[6] D. Venkatrayappa, P. Montesinos, D. Diep, and B. Magnier, "A novel image descriptor based on anisotropic filtering," in *CAIP*. Springer, 2015, pp. 161–173.

[7] J. Canny, "A computational approach to edge detection," *IEEE TPAMI*, pp. 679–698, 1986.

[8] Z. Püspöki, M. Storath, D. Sage, and M. Unser, "Transforms and operators for directional bioimage analysis: a survey," in *Focus on Bio-Image Informatics*, pp. 69–93. Springer, 2016.

[9] A Paeth, "A fast algorithm for general raster rotation. graphics gems," *AP Professional*, 1990.

[10] T.M. Lehmann, C. Gonner, and K. Spitzer, "Survey: Interpolation methods in medical image processing," *IEEE T. on Med. Im.*, vol. 18, no. 11, pp. 1049–1075, 1999.

[11] P. Thévenaz, T. Blu, and M. Unser, "Image interpolation and resampling," *Handbook of MIPA*, vol. 1, no. 1, pp. 393–420, 2000.

[12] M. Unser, P. Thévenaz, and L. Yaroslavsky, "Convolution-based interpolation for fast, high-quality rotation of images," *IEEE Transactions on image processing*, vol. 4, no. 10, pp. 1371–1381, 1995.

[13] E. Andres, "The quasi-shear rotation," in *Int. Conf. on Discr. Geom. for Comp. Im.* Springer, 1996, pp. 307–314.

[14] L. Condat and D. Van De Ville, "Fully reversible image rotation by 1-d filtering," in *IEEE ICIP*, 2008, pp. 913–916.

[15] R. Keys, "Cubic convolution interpolation for digital image processing," *IEEE TASSP*, vol. 29, no. 6, pp. 1153–1160, 1981.

[16] T. Blu, P Thévenaz, and M. Unser, "Moms: Maximal-order interpolation of minimal support," *IEEE TIP*, vol. 10, pp. 1069–1080, 2001.

[17] P. Carré, E. Andres, and C. Fernandez-Maloigne, "Discrete rotation for directional orthogonal wavelet packets," in *IEEE ICIP*, 2001, vol. 2, pp. 257–260.

[18] B. Magnier, H. Abdulrahman, and P. Montesinos, "A review of supervised edge detection evaluation methods and an objective comparison of filtering gradient computations using hysteresis thresholds," *Journal of Imaging*, vol. 4, no. 6, pp. 74, 2018.

[19] B. Magnier, "Edge detection evaluation: A new normalized figure of merit," in *IEEE ICASSP*, 2019, pp. 2407–2411.

[20] H. Abdulrahman, B. Magnier, and P. Montesinos, "From contours to ground truth: How to evaluate edge detectors by filtering," *J. of WSCG*, vol. 25, no. 2, pp. 133–142, 2017.