

optimization problem:

$$h = \arg \max_h \frac{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}, \quad (1)$$

where $[F(h)]$ denotes the Fourier transform of the $\hat{\phi}$ iter. This is similar to the aliasing energy in [5]. The difference is that we use a ratio instead. Without any constraint, the solution to (1) is the well-known *sinc* function and the energy concentration is 1. However, in practice we want the $\hat{\phi}$ iter to be $\hat{\phi}$ nitely supported. Under such a constraint, the solution to (1) becomes the prolate spheroidal wave function of order zero [9]. Unfortunately, the prolate spheroidal wave functions do not have closed-form solutions, even power-series solutions are unavailable.

For the second criterion, among all elementary functions, only polynomials are possible to have closed-form solutions to the basic component integral.

Following the above considerations, we design optimal polynomial $\hat{\phi}$ lters, either square or circular, respectively. We only present these two types of $\hat{\phi}$ lters because they are most commonly used. However, the same idea is applicable to other shapes of $\hat{\phi}$ lters.

2.1 The optimal circular polynomial $\hat{\phi}$ lters

For circular polynomial $\hat{\phi}$ lters, we may assume that $t(r, \theta) = \sum_{k=0}^M h_k r^k$, where $r \in [0, R]$ and $\theta \in [0, 2\pi)$. Then the numerator of (1) is given by

$$\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy = \sum_{k,l=0}^M h_k h_l \int_{-R}^R \int_{-R}^R r^{k+l} dx dy,$$

where

$$\int_{-R}^R \int_{-R}^R r^{k+l} dx dy = \int_{-R}^R \int_{-R}^R r^{k+l} dx dy,$$

and $\int_{-R}^R \int_{-R}^R r^m dx dy$ is the Fourier transform of r^m ($0 \leq r \leq R$):

$$\int_{-R}^R \int_{-R}^R r^m dx dy = \int_0^R r^{m+1} dr \int_0^{2\pi} e^{-ir(x \cos \theta + y \sin \theta)} d\theta = 2 \int_0^R r^{m+1} J_0(r/\rho) dr,$$

in which $J_0(\cdot)$ is the 0-th order Bessel function and $\rho = \sqrt{\frac{x^2}{2} + \frac{y^2}{2}}$.

By Parseval's theorem on Fourier transform, the denominator of (1) is given by

$$\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy = (2\pi)^2 \int_{-R}^R \int_{-R}^R |h(x, y)|^2 dx dy = (2\pi)^2 \sum_{k,l=0}^M h_k h_l \int_{-R}^R \int_{-R}^R r^{k+l} dx dy,$$

where

$$\int_{-R}^R \int_{-R}^R r^{k+l} dx dy = \int_0^R \int_0^{2\pi} r^{k+l+1} dr d\theta = \frac{2\pi R^{k+l+2}}{k+l+2}.$$

The maximization of $\frac{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}$ can be achieved by utilizing the Lagrangian multiplier:

$$\frac{\partial}{\partial h_k} \left(\frac{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy} - \lambda \right) = 0, \text{ and } \frac{\partial}{\partial \lambda} \left(\frac{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy}{\int_{-R}^R \int_{-R}^R |F(h)(x, y)|^2 dx dy} - \lambda \right) = 0. \quad (2)$$

This leads to an eigenvector problem:

$$-1 \mathbf{h} = \mathbf{h}, \quad (3)$$

$$\text{and } \mathbf{h}^T \mathbf{h} = 1, \quad (4)$$

Table 1: Some data on low-order polynomial $\hat{\mathcal{O}}$ lters.

Filter Type	Circular		Square	
Radius	1	2	1	2
h_0	0.56904256713865	0.26412404360302	0.71247167744650	0.45090000658189
h_1	0.05056692464147	-0.03455158299490	-0.79090785614255	-0.22055267318930
h_2	-0.91026906187835	-0.15934681709384	0.25582137300511	0.02911569510537
h_3	0.42672641722501	0.05631744983035		
Energy Con- centration	0.93801371604876	0.99950079154631	0.96260897769620	0.99888861744658
Maximum Energy Con- centration	0.93801385499638	0.99965857174068	0.96261252017346	0.99988655625306

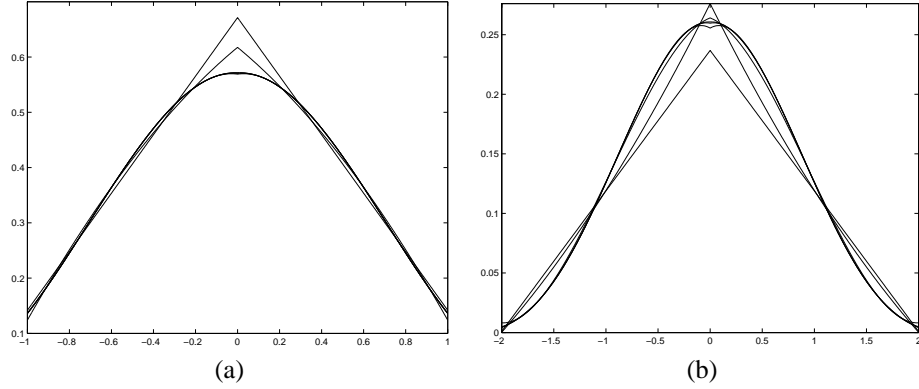


Figure 1: The central cross sections of the circular polynomial $\hat{\mathcal{O}}$ lters. (a) Filters with radius 1, where the top, middle and bottom curves correspond to $M = 1$, $M = 2$ and $M = 3$, respectively. (b) Filters with radius 2, where the bottom, top and middle curves correspond to $M = 1$, $M = 2$, and $M = 3$, respectively. Note that the curves for $M > 3$ are nearly indistinguishable from that for $M = 3$.

where $\mathbf{h} = (h_0 \ h_1 \ \dots \ h_M)^T$, $\mathbf{h} = (h_0 \ h_1 \ \dots \ h_M)^T$, and $\mathbf{h} = (h_0 \ h_1 \ \dots \ h_M)^T$. For maximization, \mathbf{h} must be chosen as the eigenvector that corresponds to the maximal eigenvalue of \mathbf{A}^{-1} and be normalized according to (4).

Though the energy concentration improves when M increases, the convergence to the maximum energy concentration is very fast, especially when $R = 2$. Because polynomials can approximate arbitrary continuous functions uniformly, the fast convergence indicates that our low-order optimal polynomial $\hat{\mathcal{O}}$ lters are actually very close to the prolate spheroidal wave function. Our numerical computation shows that $M = 3$ is enough for the corresponding $\hat{\mathcal{O}}$ lter to achieve over 99.98% of the maximum energy concentration at a given radius, achieved by the prolate spheroidal wave function. Figure 1(a) is the central cross section of the optimal circular $\hat{\mathcal{O}}$ lters with $R = 1$, where the top, middle and bottom curves correspond to $M = 1$, $M = 2$ and $M = 3$, respectively. We can see that the curves for $M > 3$ are nearly indistinguishable from that for $M = 3$. Similar results are observable in Figure 1(b), where $R = 2$. The coefficients of the optimal circular polynomial $\hat{\mathcal{O}}$ lters at different radius are listed in the left part of Table 1.

Now we give the closed-form solution to the basic component integral. For circular $\hat{\mathcal{O}}$ lters, the basic component region is shown in Figure 2(a). By defining a new coordinate, where the t -axis is along the marching direction of the polygon edge so that the interior is on the right hand side of the edge and the d -direction is $\sqrt{2}$ behind the t -direction, the basic component region can be parameterized by the coordinate of the polygon vertex V in the new coordinate, i.e., the distance d from the pixel center to the polygon edge and the distance t between the vertex V and the projection of the center onto the polygon edge. The basic component integral is $I(d, t) = \sum_{k=0}^M h_k I_k(d, t)$, where $I_k(d, t)$ is the integral of

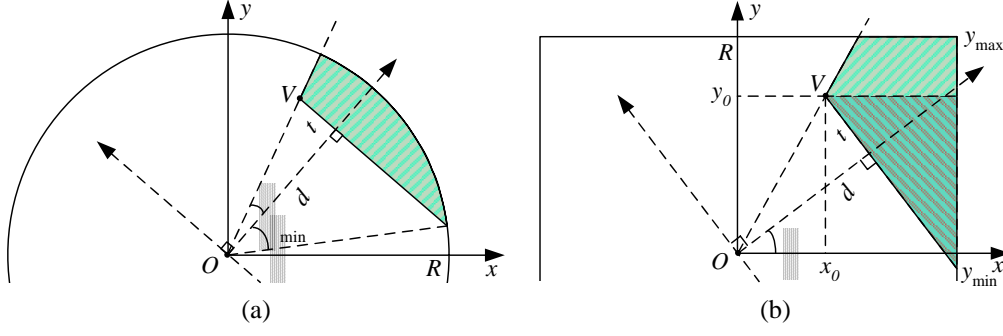


Figure 2: The basic component region (shaded area) is bounded by one polygon edge, one radius passing the polygon vertex and the filter boundary. (a) For circular filters, the parameterization (θ, t) is 2D. (b) For square filters, the parameterization (x, d, t) is 3D.

r^k over the same region. Converting to polar coordinates (Figure 2(a)), we have¹:

$$I_k(d, t) = \int_{\min}^R d \frac{r^{k+1}}{\cos \theta} dr = \frac{1}{k+2} \int_{\min}^R R^{k+2} - d^{k+2} (\cos \theta)^{-(k+2)} d \theta$$

$$= \frac{1}{k+2} R^{k+2} (\theta - \theta_{\min}) - d^{k+2} [P_{k+2}(\theta) - P_{k+2}(\theta_{\min})],$$

where

$$\theta = \arctan(t/d), \quad \theta_{\min} = -\arccos(d/R), \quad \text{and } P_k(\theta) = (\cos \theta)^{-k} d.$$

By partial integral, $P_k(\theta)$ can be computed via the following recursion:

$$P_k(\theta) = \frac{1}{k-1} (\tan \theta) (\cos \theta)^{-(k-2)} + \frac{k-2}{k-1} P_{k-2}(\theta).$$

2.2 The optimal square polynomial filters

For square polynomial filters, it is easy to see that $h(x, y)$ must be symmetric with respect to lines $x = 0$, $y = 0$, and $y = x$. Therefore, we may assume that $h(x, y) = \sum_{p=0}^M \sum_{q=0}^M h_{pq} x^{2p} y^{2q}$, with $h_{pq} = h_{qp}$, where $|x|, |y| \leq R$. Then

$$\int_{-R}^R \int_{-R}^R [F(h)](x, y)^2 dx dy = \sum_{p,q,k,l=0}^M h_{pq} h_{kl} \int_{-R}^R \int_{-R}^R x^{2p+2k} y^{2q+2l} dx dy,$$

where

$$\int_{-R}^R \int_{-R}^R x^{2m} y^{2n} dx dy = \frac{2^{2m+2n}}{(2m+1)(2n+1)} R^{2m+2n+2},$$

and $\int_{-R}^R x^{2m} dx$ is the Fourier transform of x^{2m} ($|x| \leq R$):

$$\int_{-R}^R x^{2m} e^{-ix} dx = \frac{2R^{2m+1}}{(2m+1)!} \sin(R) + \frac{2R^{2m}}{(2m)!} \cos(R) - \int_{-R}^R x^{2m-2} e^{-ix} dx.$$

By partial integral, $\int_{-R}^R x^{2m} dx$ can be computed via the following recursion:

$$\int_{-R}^R x^{2m} dx = \frac{2(R)^{2m+1}}{(2m+1)!} \sin(R) + \frac{2R^{2m}}{(2m)!} \cos(R) - \int_{-R}^R x^{2m-2} dx.$$

¹For simplicity we do not discuss all possible (d, t) , such as the case where $d > R$ or $t < -\sqrt{R^2 - d^2}$. Section 2.2 also follows this convention.

The computation of $I_{2m,2n}$ is similar, where $I_{2m,2n}$ changes to

$$I_{2m,2n} = \int_{-R}^R [F(x^{2m})] \overline{[F(x^{2n})]} dx = 2 \int_{-R}^R x^{2m} x^{2n} dx = \frac{4 R^{2(m+n)+1}}{2(m+n)+1},$$

in which Parseval's theorem is applied again.

Utilizing the Lagrangian multiplier as in (2), we have:

$$\mathbf{H} = \mathbf{H}^T, \quad (5)$$

$$\text{and } \text{trace}(\mathbf{H} \mathbf{H}^T) = 1, \quad (6)$$

where $\mathbf{H} = (h_{ij})$, $\mathbf{H} = (h_{ij})$, $\mathbf{H} = (h_{ij})$, and $\text{trace}(\mathbf{X})$ is the trace of matrix \mathbf{X} . Equation (5) can be rewritten as:

$$(\mathbf{A} - \mathbf{A}^T) \text{vec}(\mathbf{h}) = \mathbf{0} \cdot \text{vec}(\mathbf{h}),$$

where $\mathbf{A} = \mathbf{A}^{-1}$, \mathbf{A} is the Kronecker product [1] of matrices, and $\text{vec}(\mathbf{X})$ is the vectorization [1] of matrix \mathbf{X} . Let λ_{\max} be the largest eigenvalue of \mathbf{A} and $\tilde{\mathbf{h}} = (\tilde{h}_0 \tilde{h}_1 \dots \tilde{h}_M)^T$ be the corresponding eigenvector, then λ_{\max}^2 is the maximum eigenvalue of $\mathbf{A} - \mathbf{A}^T$ and $\text{vec}(\tilde{\mathbf{h}}\tilde{\mathbf{h}}^T)$ is the corresponding eigenvector ([1], more general theorem exists). Therefore, we may choose separable $\mathbf{H} = \tilde{\mathbf{h}}\tilde{\mathbf{h}}^T$, where $\tilde{\mathbf{h}}$ is normalized via $\tilde{\mathbf{h}}^T \tilde{\mathbf{h}} = 1$. This leads to a separable 2D filter

$$h(x, y) = h(x)h(y), \text{ where } h(x) = \sum_{k=0}^M \tilde{h}_k x^{2k}.$$

Again, the convergence of $h(x)$ is fast when M increases. Our numerical experiments show that $M = 2$ is sufficient for the corresponding filter $h(x, y)$ to achieve over 99.9% of the maximum energy concentration when $R = 2$. The coefficients $\tilde{h}_k(x)$ are listed in the right part of Table 1.

For square filters, the basic component region is shown in Figure 2(b), where an extra parameter, which is related to the polar sweep angle of the d -axis, is required for the parameterization. The basic component integral is $I(\theta, d, t) = \int_{-y_0}^{y_0} \int_{-x_0}^{x_0} h_{mn} I_{2m,2n}(\theta, d, t)$, where $I_{2m,2n}(\theta, d, t)$ is the integral of $x^{2m} y^{2n}$ over the basic component region. Due to symmetry, we may assume that $d \geq 0$ and $0 \leq \theta \leq \pi/4$. Breaking the basic component region into two shaded regions shown in Figure 2(b), we have

$$\begin{aligned} I_{2m,2n}(\theta, d, t) &= \int_{y_{\min}}^{y_0} \int_{-x_0}^R x^{2m} dx + \int_{y_0}^{y_{\max}} \int_{-R}^R x^{2m} dx \\ &= \frac{R^{2m+1} (y_{\max}^{2n+1} - y_{\min}^{2n+1})}{(2m+1)(2n+1)} - \frac{y_0^{2m+1} (y_{\max}^{2(m+n+1)} - y_0^{2(m+n+1)})}{2(2m+1)(m+n+1)} \\ &\quad - \frac{1}{(2m+1)(\cos \theta)^{2m+1}} \sum_{k=0}^{2m+1} (-1)^k C_{2m+1}^k (\sin \theta)^k d^{2m+1-k} \frac{y_0^{2n+k+1} - y_{\min}^{2n+k+1}}{2n+k+1}, \end{aligned}$$

where

$$\begin{aligned} y_0 &= d \sin \theta + t \cos \theta, & x_0 &= d \cos \theta - t \sin \theta, \\ y_{\min} &= \max \left\{ \frac{d - R \cos \theta}{\sin \theta}, -R \right\}, & y_{\max} &= \begin{cases} \max \{ \min \{ R / \cos \theta, R \}, -R \}, & \text{if } x_0 > 0, \\ R, & \text{if } x_0 \leq 0, \end{cases} \\ &= \frac{x_0}{\sin \theta}, & C_m^n &= \frac{m!}{n!(m-n)!}. \end{aligned}$$

3 Experimental Results

Figure 3 shows a wheel and a checkerboard rendered by our rendering system ², SplitRender [6], using the square polynomial filter with radius 2. There are 180 triangles in Figure 3(a), while Figure 3(b) has 92,100 quadrilaterals,

²The full-resolution images for Figures 3–5 are available online at <http://www.acm.org/jgt/papers/LinEtAl04>.

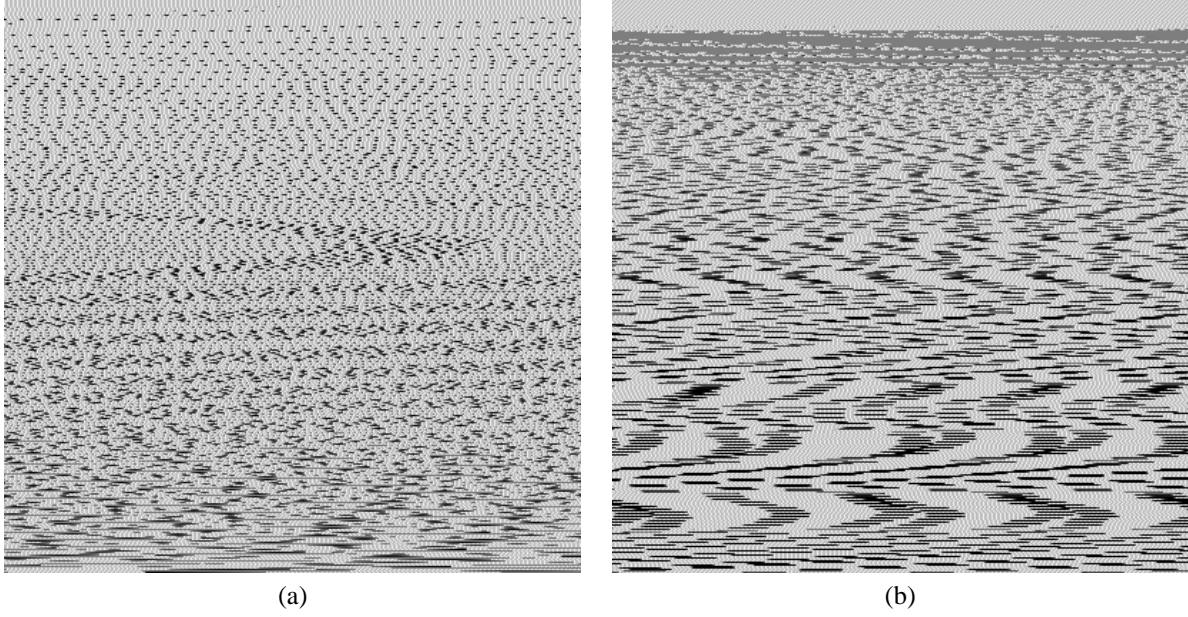


Figure 3: The rendering results of SplitRender using the square polynomial $\hat{\mathcal{O}}$ lter with radius 2, closed-form solution used.

most of which are extremely small and are at the top of the image. The rendering results are almost aliasing-free. For comparisons of our polynomial $\hat{\mathcal{O}}$ lters against various $\hat{\mathcal{O}}$ lters when pre- $\hat{\mathcal{O}}$ ltering polygons, please see [6].

Our polynomial $\hat{\mathcal{O}}$ lters can also be applied to $\hat{\mathcal{O}}$ ltering images. We now compare the anti-aliasing performance of various $\hat{\mathcal{O}}$ lters over discrete images. The test image is Figure 3(a) and will be scaled down by 1.8 times using various $\hat{\mathcal{O}}$ lters. The chosen $\hat{\mathcal{O}}$ lters include: square polynomial $\hat{\mathcal{O}}$ lter ($M=3$), circular polynomial $\hat{\mathcal{O}}$ lter ($M=3$), square Gaussian $\hat{\mathcal{O}}$ lters, circular Gaussian $\hat{\mathcal{O}}$ lters, conical $\hat{\mathcal{O}}$ lters, box $\hat{\mathcal{O}}$ lters and the Mitchell-Netravali $\hat{\mathcal{O}}$ lters [8]. The radii of the former 6 kinds of $\hat{\mathcal{O}}$ lters can be either 1 or 2, while those of the Mitchell-Netravali $\hat{\mathcal{O}}$ lters can only be 2. For Gaussian $\hat{\mathcal{O}}$ lters, we change σ from 0.1 to 1.2 to $\hat{\mathcal{O}}$ nd the best $\hat{\mathcal{O}}$ ltering results, which should have good balance between eliminating aliasing and keeping the image sharp. For Mitchell-Netravali $\hat{\mathcal{O}}$ lters, the chosen (B, C) are: $(1/3, 1/3)$, $(0, 1)$, and $(0, 0.5)$, respectively, as they have been mentioned in the literature [2, 4, 3]. The resultant images are computed via the following formula:

$$I_r(p, q) = \frac{h(\tilde{m} - p, \tilde{n} - q) I_t(m, n)}{\|(\tilde{m} - p, \tilde{n} - q)\|_R},$$

where I_t and I_r are the test image and resultant image, respectively, (\tilde{m}, \tilde{n}) is the coordinate of pixel (m, n) of image I_t in image I_r , and

$$\|(x, y)\| = \begin{cases} \sqrt{x^2 + y^2}, & \text{if the } \hat{\mathcal{O}} \text{ lter is circular,} \\ \max\{|x|, |y|\}, & \text{if the } \hat{\mathcal{O}} \text{ lter is square.} \end{cases}$$

Figures 4 are closeups of the down-scaling results of $\hat{\mathcal{O}}$ lters with radius 1, where $\sigma = 0.3$ for Gaussian $\hat{\mathcal{O}}$ lters. We see that square and circular polynomial $\hat{\mathcal{O}}$ lters (Figures 4(a) and (b)) result in least aliasing. The performance of square Gaussian (Figure 4(c)), circular Gaussian (Figure 4(d)) and conical $\hat{\mathcal{O}}$ lters (Figure 4(e)) is very close, and the box $\hat{\mathcal{O}}$ lter (Figure 4(f)) is the worst.

Figures 5 are closeups of the down-scaling results of $\hat{\mathcal{O}}$ lters with radius 2, where $\sigma = 0.9$ for Gaussian $\hat{\mathcal{O}}$ lters. We see that square and circular polynomial $\hat{\mathcal{O}}$ lters and square Gaussian $\hat{\mathcal{O}}$ lters (Figures 5(a)) have the least aliasing. Circular Gaussian (Figure 5(d)) and conical $\hat{\mathcal{O}}$ lters (Figure 5(e)) come next, and again the box $\hat{\mathcal{O}}$ lter (Figures 5(f)) is the worst. The results of Mitchell-Netravali $\hat{\mathcal{O}}$ lters (Figures 5(g)(i)) are very close to each other. They look sharper than the other images but their Moiré patterns are also more severe. It is interesting to compare them with our polynomial $\hat{\mathcal{O}}$ lters with

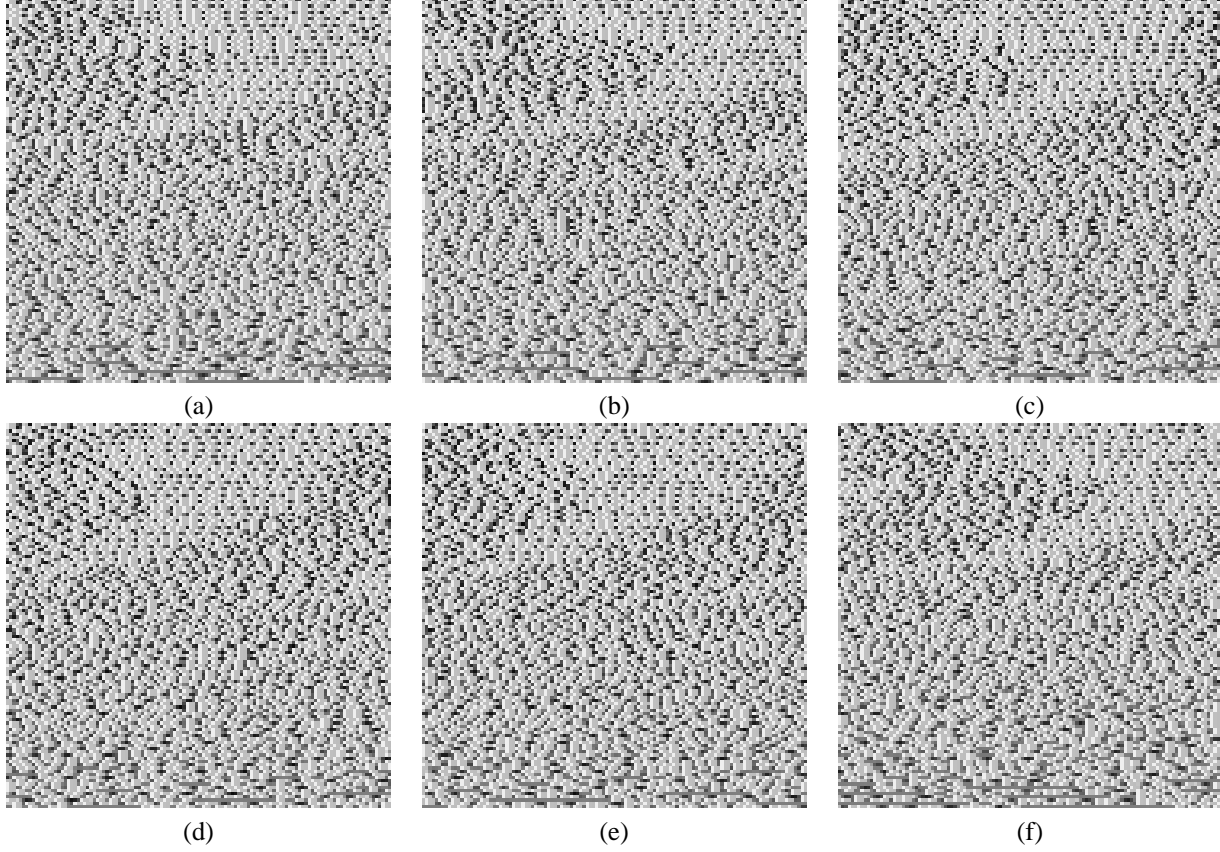


Figure 4: Closeups of downsampling Figure 3(a) by 1.8 times using various \hat{O} lters with radius 1. (a) Square polynomial \hat{O} lter. (b) Circular polynomial \hat{O} lter. (c) Square Gaussian \hat{O} lter with $\sigma = 3$. (d) Circular Gaussian \hat{O} lter with $\sigma = 0.3$. (e) Conical \hat{O} lter. (f) Box \hat{O} lter.

radius 1 (Figures 4(a) and (b)). Figures 4(a) and (b) are as sharp as Figures 5(g)–(i), but their aliasing effect is slightly less severe.

From the above comparisons, and from the polygon- \hat{O} ltering comparisons in [6], we may draw the following conclusions:

1. Our low-order optimal polynomial \hat{O} lters do have excellent anti-aliasing performance.
2. The anti-aliasing performance of our polynomial \hat{O} lters with radius 1 is comparable with that of the Mitchell-Netravali \hat{O} lters, which has a radius 2 and negative lobes. This makes our polynomial \hat{O} lters with radius 1 very useful because, without sacrificing performance, smaller radius often saves computation and at the same time its non-negativity can avoid the problems of clipping and ringing artifact that may result from the negative lobes of other \hat{O} lters.
3. Filters using closed-form evaluation are more suitable for high-quality anti-aliasing. Using look-up tables or super-sampling always introduces random noise if the sizes of graphical objects are beyond their precision.³

References

- [1] Yun-Peng Cheng, Kai-Yuan Zhang, and Zhong Xu. *Matrix Theory (in Chinese)*. Northwest Industrial University Press, Xi'an, Shaanxi, China, 2000.

³This is a conclusion in [6].

- [2] Tom Duff. Polygon scan conversion by exact convolution. *Raster Imaging and Digital Typography* 89, pages 154–168, 1989.
- [3] A. E. Fabris and A. R. Forrest. Antialiasing of curves by discrete pre-filtering. *SIGGRAPH 1997 Conference Proceedings*, Annual Conference Series, pages 317–323, August 1997.
- [4] Brian Guenter and Jack Tumblin. Quadrature pre-filtering for high quality antialiasing. *ACM Trans. on Graphics*, 15(4):332–353, 1996.
- [5] J. Kajiya and M. Ullner. Filtering high quality text for display on raster scan devices. In *SIGGRAPH 1981 Conference Proceedings*, Annual Conference Series, pages 7–15, August 1981.
- [6] Zhouchen Lin, Hai-Tao Chen, Heung-Yeung Shum, and Jian Wang. Pre-filtering 2d polygons without clipping. *Journal of Graphics Tools*, 9(1):1–11, 2004.
- [7] M. D. McCool. Analytic antialiasing with prism splines. In *SIGGRAPH 1995 Conference Proceedings*, Annual Conference Series, pages 429–436, August 1995.
- [8] Don P. Mitchell and Arun N. Netravali. Reconstruction filters in computer graphics. *SIGGRAPH 1988 Conference Proceedings*, Annual Conference Series, pages 221–227, August 1988.
- [9] D. Slepian and H. O. Pollak. Prolate spheroidal wave functions, fourier analysis and uncertainty. 1. *The Bell System Technical Journal*, 40:43–64, 1961.

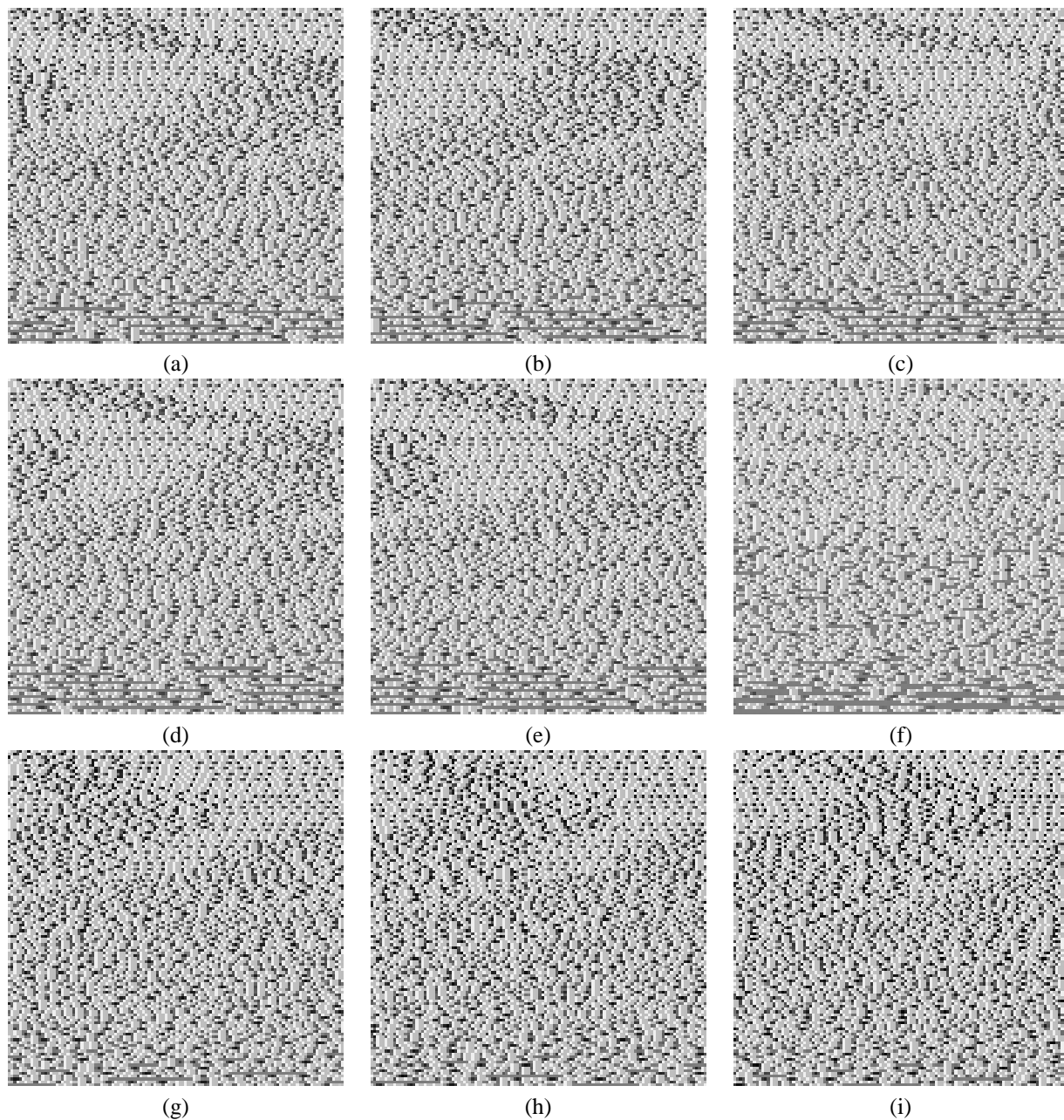


Figure 5: Closeups of downsampling Figure 3(a) by 1.8 times using various $\hat{\phi}$ lters with radius 2. (a) Square polynomial $\hat{\phi}$ lter. (b) Circular polynomial $\hat{\phi}$ lter. (c) Square Gaussian $\hat{\phi}$ lter with $\sigma = 0.9$. (d) Circular Gaussian $\hat{\phi}$ lter with $\sigma = 0.9$. (e) Conical $\hat{\phi}$ lter. (f) Box $\hat{\phi}$ lter. (g) Mitchell-Netravali $\hat{\phi}$ lter with $B = 1/3$, $C = 1/3$. (h) Mitchell-Netravali $\hat{\phi}$ lter with $B = 0$, $C = 0.5$. (i) Mitchell-Netravali $\hat{\phi}$ lter with $B = 0$, $C = 1$.