

Survey: Interpolation Methods in Medical Image Processing

Thomas M. Lehmann,* *Member, IEEE*, Claudia Gönner, and Klaus Spitzer

Abstract—Image interpolation techniques often are required in medical imaging for image generation (e.g., discrete back projection for inverse Radon transform) and processing such as compression or resampling. Since the ideal interpolation function spatially is unlimited, several interpolation kernels of finite size have been introduced. This paper compares 1) truncated and windowed sinc; 2) nearest neighbor; 3) linear; 4) quadratic; 5) cubic B-spline; 6) cubic; g) Lagrange; and 7) Gaussian interpolation and approximation techniques with kernel sizes from 1×1 up to 8×8 . The comparison is done by: 1) spatial and Fourier analyses; 2) computational complexity as well as runtime evaluations; and 3) qualitative and quantitative interpolation error determinations for particular interpolation tasks which were taken from common situations in medical image processing.

For local and Fourier analyses, a standardized notation is introduced and fundamental properties of interpolators are derived. Successful methods should be direct current (DC)-constant and interpolators rather than DC-inconstant or approximators. Each method's parameters are tuned with respect to those properties. This results in three novel kernels, which are introduced in this paper and proven to be within the best choices for medical image interpolation: the 6×6 Blackman–Harris windowed sinc interpolator, and the C2-continuous cubic kernels with $N = 6$ and $N = 8$ supporting points.

For quantitative error evaluations, a set of 50 direct digital X rays was used. They have been selected arbitrarily from clinical routine. In general, large kernel sizes were found to be superior to small interpolation masks. Except for truncated sinc interpolators, all kernels with $N = 6$ or larger sizes perform significantly better than $N = 2$ or $N = 3$ point methods ($p \ll 0.005$). However, the differences within the group of large-sized kernels were not significant. Summarizing the results, the cubic 6×6 interpolator with continuous second derivatives, as defined in (24), can be recommended for most common interpolation tasks. It appears to be the fastest six-point kernel to implement computationally. It provides eminent local and Fourier properties, is easy to implement, and has only small errors. The same characteristics apply to B-spline interpolation, but the 6×6 cubic avoids the intrinsic border effects produced by the B-spline technique.

However, the goal of this study was not to determine an overall best method, but to present a comprehensive catalogue of methods in a uniform terminology, to define general properties and requirements of local techniques, and to enable the reader to select that method which is optimal for his specific application in medical imaging.

Index Terms—Approximation, B-splines, cubic polynomials, image resampling, interpolation.

I. INTRODUCTION

IMAGE interpolation has many applications in computer vision. It is the first of the two basic resampling steps and transforms a discrete matrix into a continuous image. Subsequent sampling of this intermediate result produces the resampled discrete image. Resampling is required for discrete image manipulations, such as geometric alignment and registration, to improve image quality on display devices or in the field of lossy image compression wherein some pixels or some frames are discarded during the encoding process and must be regenerated from the remaining information for decoding. Therefore, image interpolation methods have occupied a peculiar position in medical image processing [1]. They are required for image generation as well as in image post-processing. In computed tomography (CT) or magnetic resonance imaging (MRI), image reconstruction requires interpolation to approximate the discrete functions to be back projected for inverse Radon transform. In modern X-ray imaging systems such as digital subtraction angiography (DSA), interpolation is used to enable the computer-assisted alignment of the current radiograph and the mask image. Moreover, zooming or rotating medical images after their acquisition often is used in diagnosis and treatment, and interpolation methods are incorporated into systems for computer aided diagnosis (CAD), computer assisted surgery (CAS), and picture archiving and communication systems (PACS).

Image interpolation methods are as old as computer graphics and image processing. In the early years, simple algorithms, such as nearest neighbor or linear interpolation, were used for resampling. As a result of information theory introduced by Shannon in the late 1940's, the sinc function was accepted as the interpolation function of choice. However, this ideal interpolator has an infinite impulse response (IIR) and is not suitable for local interpolation with finite impulse response (FIR). From the mathematical point of view, Taylor or Lagrange polynomials have been suggested to approximate the sinc function [2]. This is documented in most textbooks on numerical analysis [3]. Thereafter, due to their numerical efficiency, different families of spline functions have been used instead.

A great variety of methods with confusing naming can be found in the literature of the 1970's and 1980's. B-splines sometimes are referred to as cubic splines [4], while cubic

Manuscript received August 6, 1998; revised September 24, 1999. This work was supported in part by the German Research Community (DFG) under Grants Re 427/5-1, Sp 538/1-2, and Schm 1268/1-2. The Associate Editor responsible for coordinating the review of this paper and recommending its publication was A. Manduca. Asterisk indicates corresponding author.

*T. M. Lehmann, C. Gönner, and K. Spitzer are with the Institute of Medical Informatics, Aachen University of Technology (RWTH), 52057 Aachen, Germany (e-mail: lehmann@computer.org).

Publisher Item Identifier S 0278-0062(99)10280-5.

TABLE I
PREVIOUS PAPERS COMPARING MORE THAN THREE INTERPOLATION METHODS

Interpolation scheme	[4]	[5]	[6]	[7]	[8]	[10]	[11]	[12]	[13]	[14]
Truncated sinc	Ac							AC	AB	
Windowed sinc								ABC	ABc	
Nearest neighbor	Ac	AB		AB		Ac	ABc	ABC	ABc	acD
Linear	Ac	ABc	AB	ABc	acC	ABc	ABc	ABC	ABc	aBcCD
Quadratic (approx.)							ABc			
Quadratic (interpo.)							ABc			
B-spline (approx.)	ABc	A		AB		ABc	ABc		A	ad
B-spline (interpol.)	abc		AB			ABcd			ac	aBcCD
Cubic, 2×2			A		acC			A		
Cubic, 4×4		ABcd	AB	ABc	acC		ABc	AC	ABc	aBcCD
Cubic, 6×6		ad								
Cubic, 8×8					acC					
Lagrange								ABC		
Gaussian										

Abbreviations: (a) kernels' derivation, (A) including plots; (b) Fourier analysis, (B) including plots; (c) image based qualitative comparison by subjects and (C) quantitative interpolation error determination; (d) complexity evaluation and (D) runtime measurements. Note that this paper covers the cubic 2-point and 8-point methods with 'abcCdD' and all other rows with 'ABcCdD'.

interpolation is also known as cubic convolution [5], [6], high-resolution spline interpolation [7], and bi-cubic spline interpolation [8], [9]. In 1983, Parker, Kenyon, and Troxel published the first paper entitled "Comparison of Interpolation Methods" [7], followed by a similar study presented by Mealand in 1988 [6]. However, previous work of Hou and Andrews, as well as that of Keys, also compare global and local interpolation methods ([4] and [5], respectively) (Table I). The Fourier transform was used in these studies to evaluate different 2×2 and 4×4 interpolation methods. Parker *et al.* pointed out that, at the expense of some increase in computing time, the quality of resampled images can be improved using cubic interpolation when compared to nearest neighbor, linear, or B-spline interpolation. However, to avoid further perpetuation of misconceptions, which have appeared repeatedly in the literature, it might be better to refer to their B-spline technique as B-spline approximation instead of interpolation. Maeland named the correct (natural) spline interpolation as B-spline interpolation and found this technique to be superior to cubic interpolation [6].

In more recent reports, not only hardware implementations for linear interpolation [15] and fast algorithms for B-spline interpolation [10] or special geometric transforms [8], [9], [14], but also nonlinear and adaptive algorithms for image zooming with perceptual edge enhancement [16], [17] have been published. However, smoothing effects are most bothersome if large magnifications are required. In addition, shape-based and object-based methods have been established in medicine for slice interpolation of three-dimensional (3-D) data sets [18]. In 1996, Appledorn presented a new approach to the interpolation of sampled data [19]. His interpolation functions are generated from a linear sum of a Gaussian function and their even derivatives. Kernel sizes of 8×8 were suggested. Again, Fourier analysis was used to optimize the parameters of the interpolation kernels. Contrary to large kernel sizes and complex interpolation families causing a high amount of computation, the use of quadratic polynomials on small regions was recommended by Dodgson in 1997. He reduces

the computation time of cubic kernels to 60% by the use of quadratic functions yielding similar quality [11].

Table I summarizes previous work comparing interpolation methods. In addition to Appledorn's and Dodgson's recent proposals, most comparative studies include neither the windowed sinc technique nor the Lagrange method and also exclude large kernels for cubic interpolation with 6×6 or 8×8 supporting points. We will see below that those methods perform superiorly in most applications.

In medical diagnostic applications, not only the kernel's frequency properties must be taken into account but also the appearance of images after resampling. Keep in mind that many imaging systems violate the sampling theorem and introduce aliasing. Unser, Aldroubi, and Eden asked subjects to rank as lifelike the magnified Lena test image in descending order [10]. Although this type of evaluation particularly depends on the images and their geometric transforms, visually assessed interpolation quality was found to be important for kernel selection [4], [5], [7], [11], [13], [14], [16], [20]. Others use the Fourier power spectrum of their test images before and after interpolation to determine the quality of their technique [8], [17]. Alternatively, Schaum suggests an error spectrum for comparing the performance of various interpolation methods [12]. Table I also summarizes characteristics used by previous authors to compare interpolation methods.

However, other bases for evaluating the appropriateness of a given interpolation scheme may be indicated, depending on the task. For example, the visual performance can be quantified in terms of similarity or sharpness [17], [21]. Furthermore, magnifications by factors of eight [10] or even two seldom are required in many clinical applications and, thus, comparing this capability may be irrelevant if not inappropriate for certain tasks. Accordingly, determining the optimal interpolation method for a variety of specific clinical applications is still a problem.

This paper presents a comprehensive survey of existing image-interpolation methods. They are expressed using a standardized terminology and are compared by means of lo-

cal and Fourier analyses, qualitative and quantitative error determinations, computational complexity evaluations, and run time measurements. Three representative interpolation tasks have been selected from clinical routine for comparing performance. These are described in detail in the next section. Relative performance and task-specific dependencies likewise are examined with regard to variations between image types and transform parameters. The interpolation methods as well as their parameters and variations are presented and discussed in Section III. The results of analyzing Fourier properties, interpolation errors, and run time are presented in Section IV and discussed in the last section of this paper.

II. INTERPOLATION TASKS IN MEDICAL IMAGING

Image resampling is required for every geometric transform of discrete images except shifts over integer distances or rotations about multiples of 90° . Geometric transforms differ with respect to their complexity. Usually, affine transforms such as magnifications [4], [7], [10], [11], [16], [17], [20], [22] or rotations [1], [8], [9], [14] are used to evaluate interpolation methods. High enlargements up to 4; 5.25; and 8 times ([11], [16], [22]; [5]; and [4], [10], [22], respectively) as well as 10 or 16 repetitions of the transform ([20] and [14] respectively) are used to enhance the blurring effects incorporated with interpolation. In this paper, simple expansions in one dimension, different rotations, and complex perspective transforms within the range of clinical applications are used to compare the interpolation methods. Representative clinical images produced by a variety of diagnostic modalities provide a reasonable basis for evaluation. These include medical photographs, magnetic resonance displays, and digital radiographs.

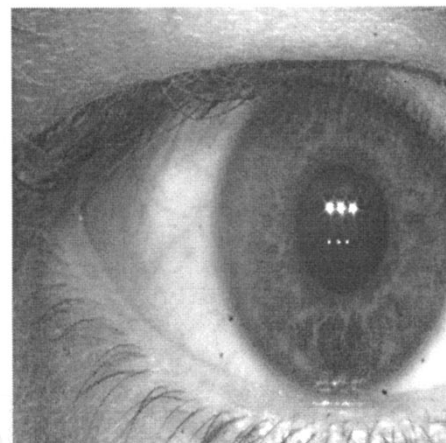
A. Correction of Aspect Ratios in CCD-Photographs

Magnifications in only one dimension are required to correct aspect ratios of digital photographs acquired with CCD-sensors and frame grabber cards. To preserve the original information, one dimension of the image is expanded, rather than shrinking the other.

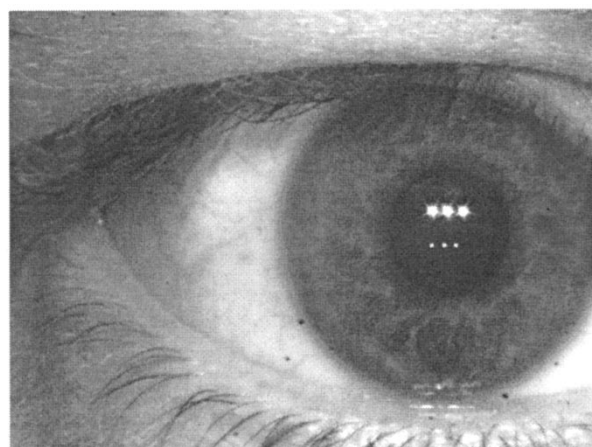
Fig. 1 shows a digital photograph of a human eye. The positions of the Purkinje reflections within the pupil are used for strabometry [23]. The picture was selected because of its uniform histogram and its excellent sharpness. The defraction patterns of the reflections in the pupil and the eyelashes are particularly sharp and thus provide an interpolation challenge. The correction of its $4/3$ -aspect ratio requires the expansion of the x axis by $1.\bar{3}$. Note that in this task the pixels in every third column of the expanded image can be taken from the original data without any modification.

B. Rotation of MRI Sections

Fig. 2 shows an MRI of the human head. In multimodal image registration, CT or MRI slices must be transformed as an affine projection to fit data from functional imaging modalities, such as positron emission tomography (PET) or single photon emission computed tomography (SPECT) [24]. By restricting affine transformations to rotations, the number of image points remains approximately constant both before and



(a)



(b)

Fig. 1. CCD-arrays or frame grabber often require the correction of their aspect ratio. (a) In this example, the image of a human eye was acquired for strabometry [23]. (b) The $4/3$ expansion in x direction was performed by linear interpolation.

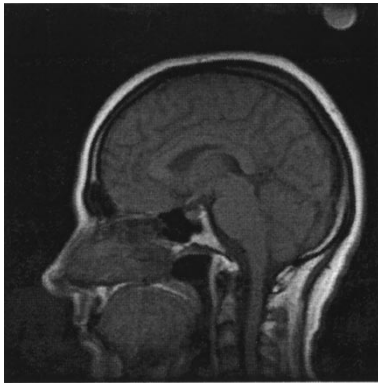
after resampling. In performing rotations on a discrete grid, neither image rows nor columns can be reproduced unless the rotation angle is a multiple of 90° . Therefore, this task results in larger interpolation errors than those produced by a simple correction of aspect ratio.

C. Perspective Projection of X-Ray Images

In intraoral radiology, the geometric transform of one radiograph into another from the same dental region acquired at a different time is described by a perspective projection [25]. Each pixel (x, y) within the reference image is transformed into the position (x', y') in the subsequent image

$$x' = \frac{a_1x + a_2y + a_3}{a_7x + a_8y + 1} \quad \text{and} \quad y' = \frac{a_4x + a_5y + a_6}{a_7x + a_8y + 1}. \quad (1)$$

Suppose Fig. 3(a) was acquired with an imaging plate perpendicular to the beam cone and the viewpoint is behind the image plate looking toward the X-ray tube. Fig. 3(b) illustrates the rotation of the image plate with its upper right corner moving toward the X-ray tube. Although distortions in clinical routine are less drastic, the parameters a_i of the perspective



(a)



(b)

Fig. 2. Interpolation is required to rotate discrete images. The MR image (a) was 45° rotated using linear interpolation (b).

projection have been chosen such that the number of pixels containing information is roughly halved after the projection. Therefore, the interpolation errors will be greatest within the examples presented here.

III. INTERPOLATION METHODS

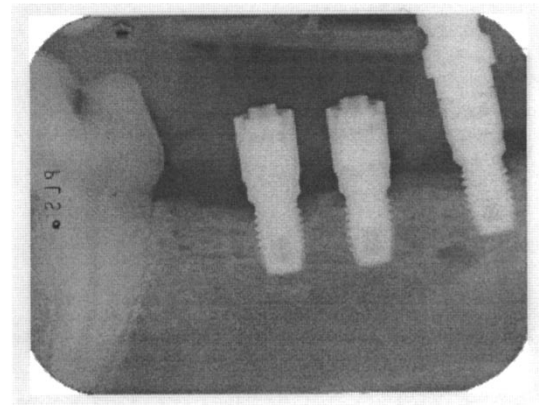
For image resampling, the interpolation step must reconstruct a two-dimensional (2-D) continuous signal $s(x, y)$ from its discrete samples $s(k, l)$ with $s, x, y \in \mathbb{R}$ and $k, l \in \mathbb{N}^0$. Thus, the amplitude at the position (x, y) must be estimated from its discrete neighbors. This can be described formally as the convolution of the discrete image samples with the continuous 2-D impulse response ${}_{2D}h(x, y)$ of a 2-D reconstruction filter

$$s(x, y) = \sum_k \sum_l s(k, l) \cdot {}_{2D}h(x - k, y - l). \quad (2)$$

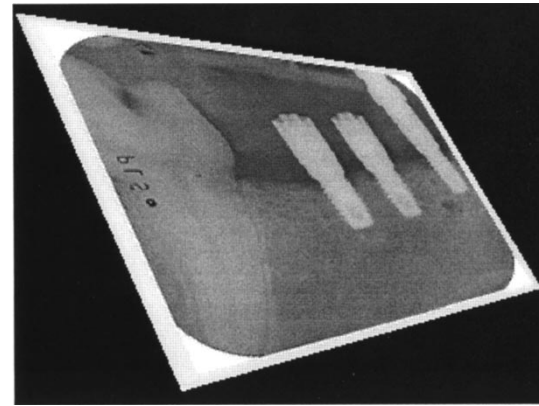
Usually, symmetrical and separable interpolation kernels are used to reduce the computational complexity

$${}_{2D}h(x, y) = h(x) \cdot h(y). \quad (3)$$

Fig. 4 illustrates the interpolation of the point (x, y) in a 4×4 neighborhood. Interpolation is performed in the x direction



(a)



(b)

Fig. 3. Intraoral X rays are acquired with respect to perspective projection. Therefore, perspective transforms must be performed for automatic adjustment. The parameters a_i in (1) are chosen as follows: $a_1 = 1.25$, $a_2 = 0.35$, $a_3 = 1.10$, $a_4 = 0.20$, $a_5 = 0.80$, $a_6 = 5.00$, $a_7 = 0.002$, $a_8 = -0.0006$. This equals a movement of the upper right corner of the image plate toward the X-ray tube. The 340×256 original pixels (a) are reduced to only 39 770 pixels (45.7%) which still contain image information after interpolation (b).

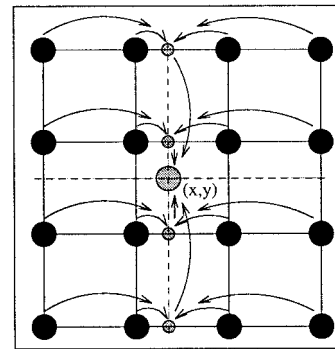


Fig. 4. One-dimensional decomposition of the 2-D $N \times N$ interpolation of the point (x, y) . (Reprinted with permission from [1].)

first. The small grey intermediate points in Fig. 4 are generated by four one-dimensional (1-D) interpolations. They are used for the final 1-D interpolation in the y direction.

A. Ideal Interpolation

Following the sampling theory, the scanning of a continuous image $s(x, y)$ yields infinite repetitions of its continuous

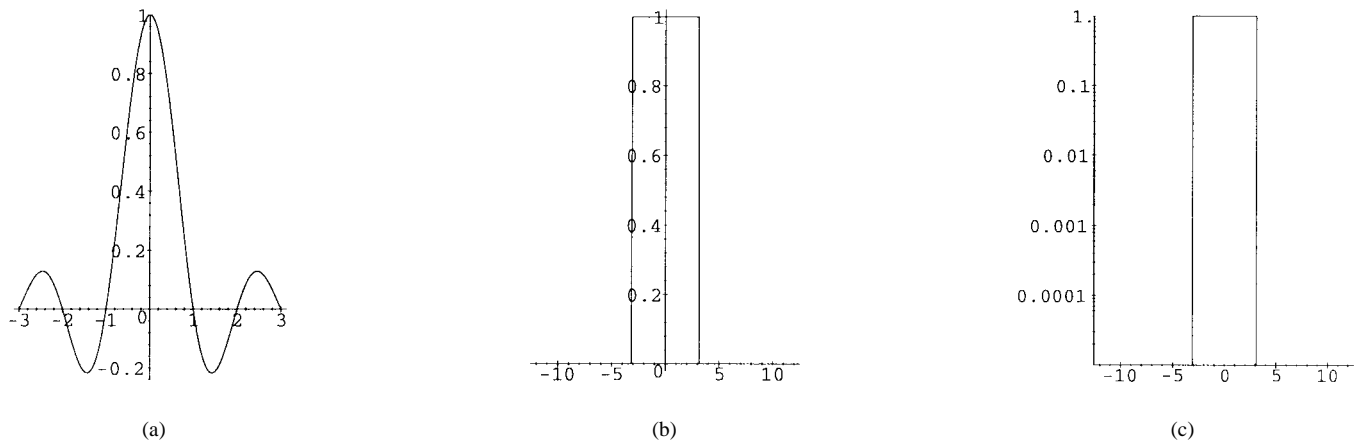


Fig. 5. Ideal interpolation. (a) Kernel plotted for $|x| < 3$. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

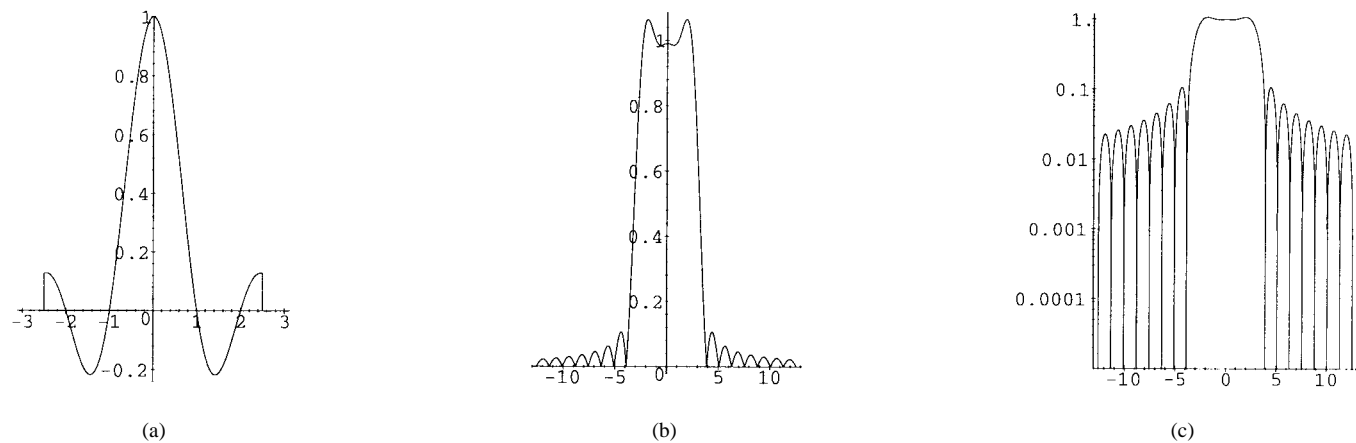


Fig. 6. Truncated sinc interpolation, $N = 5$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

spectrum $S(u, v)$ in the Fourier domain, which do not overlap since the Nyquist criterion is satisfied. If this is so, and only then, the original image $s(x, y)$ can be reconstructed perfectly from its samples $s(k, l)$ by multiplication of an appropriate rectangular prism in the Fourier domain. The 1-D ideal interpolation equals the multiplication with a rect function in the Fourier domain and can be realized in the spatial domain by a convolution with the sinc function

$$\text{Ideal}h(x) = \frac{\sin(\pi x)}{\pi x} = \text{sinc}(x). \quad (4)$$

Fig. 5(a) shows the ideal IIR-interpolator $\text{Ideal}h(x)$. The plot was truncated within the interval $-3 < x < 3$. The magnitude $|\text{Ideal}H(\omega)|$ of the Fourier transform $\text{Ideal}H(\omega)$ of the infinite kernel $\text{Ideal}h(x)$ is plotted within the interval $-4\pi \leq \omega = 2\pi f \leq 4\pi$ is shown in Fig. 5(b). The interval $-\pi < \omega < \pi$ is called passband and $f = 1/2$ or $\omega = \pi$ the cutoff point or Nyquist frequency. The transfer function of the ideal interpolator is constant and one in the passband. In addition, a logarithmical plot of the filter's Fourier response is presented in Fig. 5(c) to emphasize ripples in the stopband $|\omega| > \pi$. The ideal kernel's transfer function is zero valued within the stopband. Note that Figs. 5–24 are constructed in similar fashion using the same scaling and plotting conventions.

Some fundamental properties of any interpolator can be derived from this ideal interpolation function. $\text{Ideal}h(x)$ is positive from zero to one, negative from one to two, positive from two to three, and so on. For $h(0) \equiv 1$ these zero crossings guarantee that the image is not modified if it is resampled on the same grid. Therefore, kernels satisfying

$$\begin{cases} h(0) \equiv 1 \\ h(x) \equiv 0, \quad |x| = 1, 2, \dots \end{cases} \quad (5)$$

avoid smoothing and preserve high frequencies. They are called interpolators. We will see below that better suited kernel functions tend to have this general shape. In contrast to interpolators, kernels that do not fulfill (5) are named approximators. Note that this strict distinction is not always reflected in literature.

Sampling the interpolated (continuous) image is equivalent to interpolating the (discrete) image with a sampled interpolation function [7]. The sampling of the interpolation function aliases the higher frequencies of the interpolation function into the lower ones. In the case of ideal interpolation only, higher frequencies do not exist and therefore, within the interval $-0.5 < f < 0.5$, the sampled interpolation function has the same Fourier spectrum as the unsampled function. However, it is necessary to examine not only the continuous interpolation

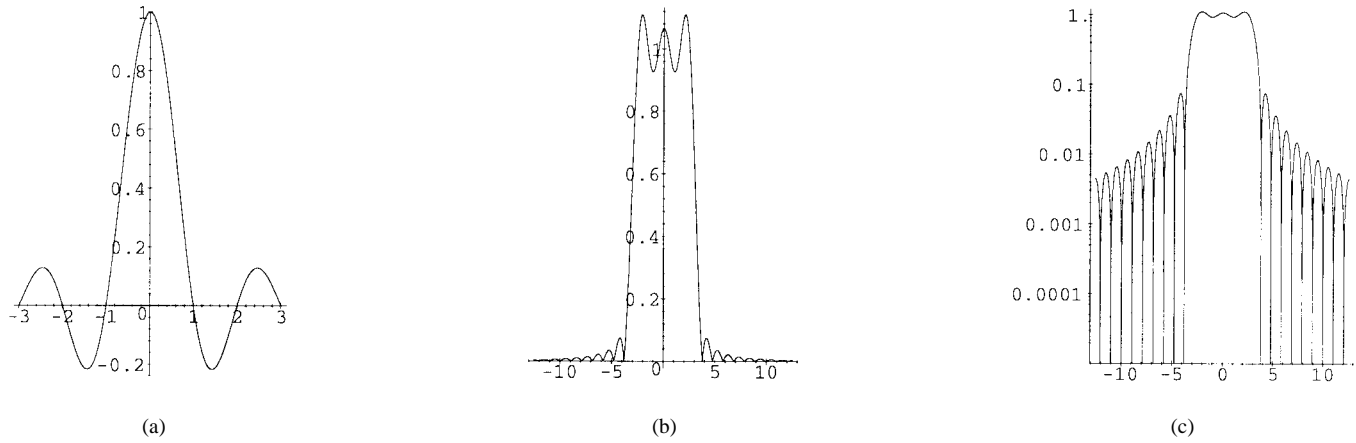


Fig. 7. Truncated sinc interpolation, $N = 6$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

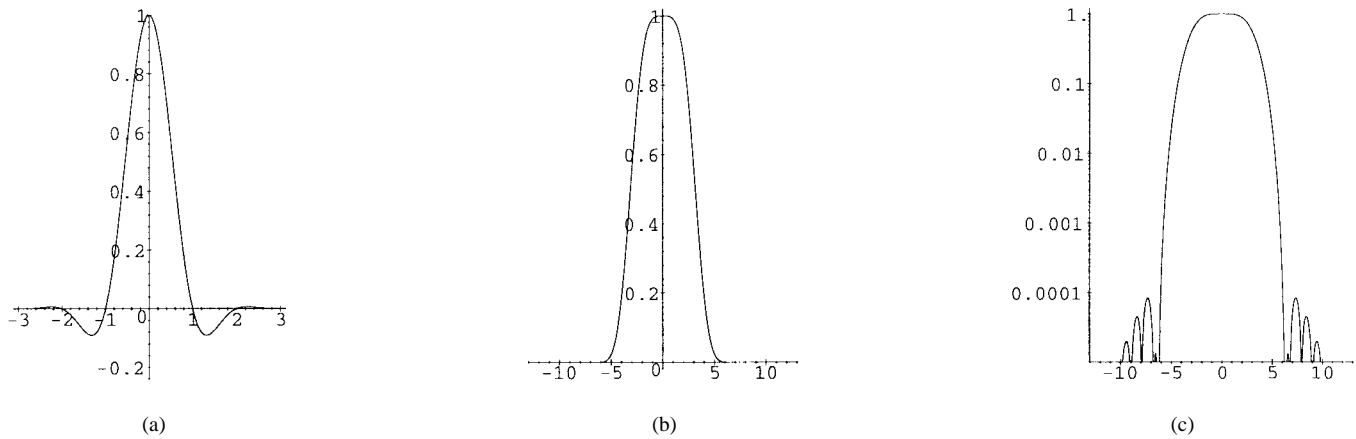


Fig. 8. Blackman-Harris windowed sinc interpolation $N = 6$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

function $h(x)$ but also typically sampled interpolation functions $h(k)$. Particularly, the sum of all samples should be one for any displacement $0 \leq d < 1$

$$\sum_{k=-\infty}^{\infty} h(d+k) \equiv 1. \quad (6)$$

This means that for any displacement d the direct current (DC)-amplification will be unity and the energy of the resampled image remains unchanged. In other words, the mean brightness of the image is not affected if the image is interpolated or resampled. Therefore, kernel functions that satisfy or fail condition (6) are named DC-constant or DC-inconstant, respectively. The next sections will show that superior kernels are DC-constant.

Equation (6) also is called the partition of unity condition [20], which easily can be evaluated in the Fourier domain. Referring to information theory, the sum (6) of discrete samples of the kernel $h(x)$ equals the area under the continuous function obtained by multiplying (or sampling) $h(x)$ with a train of delta functions $\delta(x)$

$$\sum_{k=-\infty}^{\infty} h(d+k) = \int_{-\infty}^{\infty} h(x) \cdot \text{III}(x+d) dx \quad (7)$$

where

$$\text{III}(x) = \sum_{k=-\infty}^{\infty} \delta(x+k)$$

and, by definition, the weight of a single delta impulse corresponds to the amplitude of the kernel h at the position of δ . Recognizing the Russian letter “scha,” the train of delta functions is named the scha-function $\text{III}(x)$. If the integrand in (7) is extended by $e^{-i2\pi fx}$, which equals 1 for $f = 0$, one can discover the definition of the Fourier transform in (7). Then, the function to be Fourier transformed is $h(x) \cdot \text{III}(x+d)$ and from (6) and (7) we obtain

$$H(f) * \text{III}(f) \cdot e^{i2\pi fd} \big|_{f=0} \equiv 1 \Leftrightarrow \begin{cases} H(0) \equiv 1 \\ H(f) \equiv 0, & |f| = 1, 2, \dots \end{cases} \quad (8)$$

where $H(f)$ denotes the Fourier transform of $h(x)$. Because the conditions in (8) are not sufficient but necessary in the context of interpolation, they are used to distinguish DC-constant from DC-inconstant kernels in the Fourier domain.

B. Sinc Interpolation

Although the sinc function provides an exact reconstruction of $s(x, y)$, it spatially is unlimited. There are two common

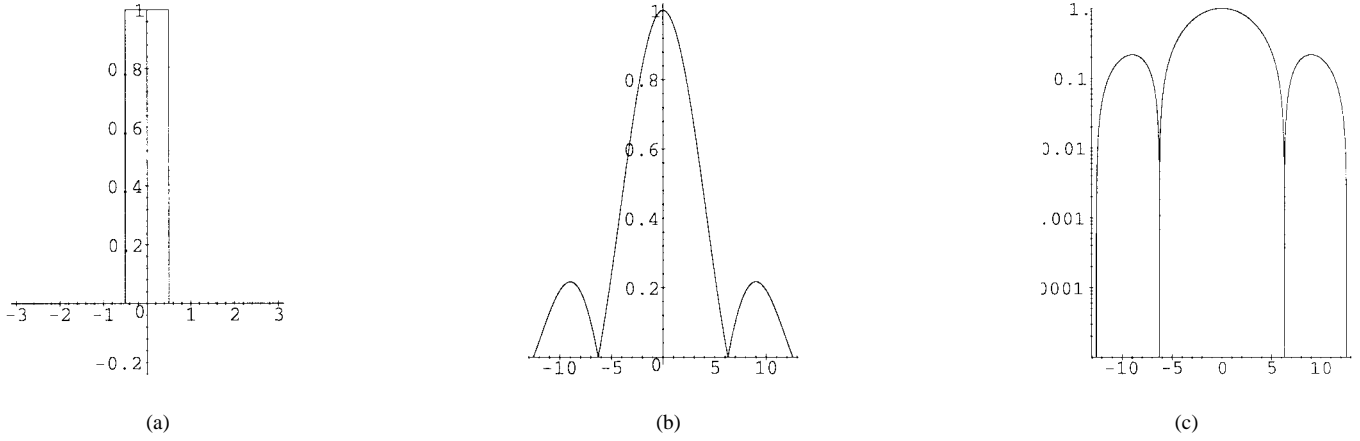


Fig. 9. Nearest neighbor interpolation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

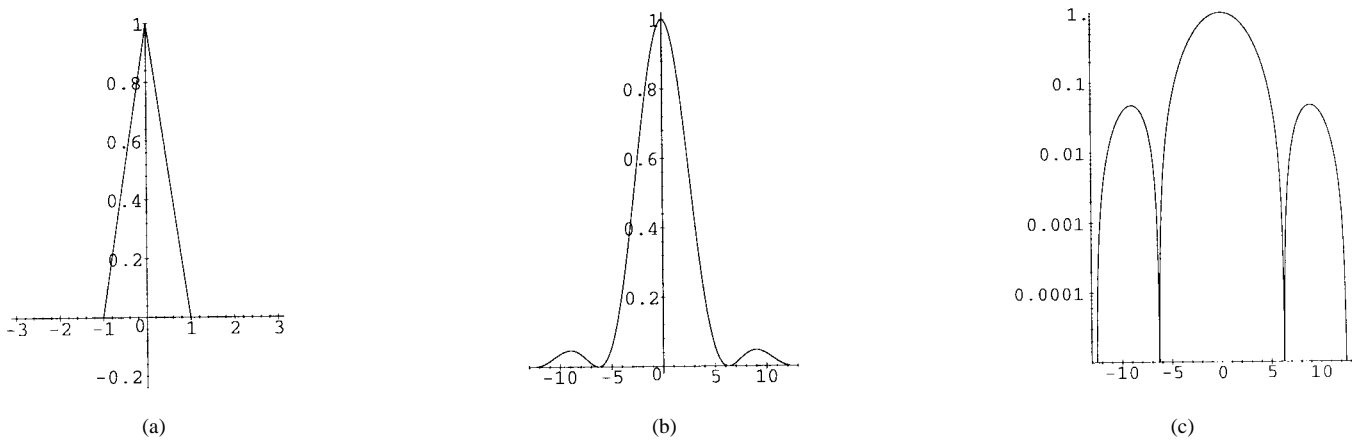


Fig. 10. Linear interpolation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

approaches for overcoming this drawback, truncation and windowing with a window function $w(x) = \text{const}(x) = 1$ and $w(x) \neq \text{const}(x)$, respectively

$$\text{Sinc} h_N(x) = \begin{cases} \text{Ideal} h(x) \cdot w(x), & 0 \leq |x| < N/2 \\ 0, & \text{elsewhere} \end{cases} \quad (9)$$

where N denotes the number of the finite kernel's supporting points. By definition, $\text{Sinc} h_N(x)$ fulfills the requirement (5). In other words, all windowed or truncated sinc kernels necessarily are real interpolators.

Truncation is equivalent to the multiplication of $\text{Ideal} h(x)$ with a rectangular function in the spatial domain, which is tantamount to a convolution with a sinc function in the frequency domain. Therefore, truncations of the ideal interpolator produce ringing effects in the frequency domain because a considerable amount of energy is discarded. Figs. 6 and 7 demonstrate this effect, which also is referred to as the Gibbs's phenomenon [4], produced by a truncated sinc function with $N = 5$ and $N = 6$ supporting points, respectively. In addition, the partition of unity condition (8) is violated by any choice of $N < \infty$. In other words, all truncated sinc kernels are DC-inconstant. The area of the function differs more from one for even kernel sizes than for odd. Therefore, raising the kernel from $N = 5$ to $N = 6$ significantly enlarges the overshoots

within the passband. With respect to the passband properties of a truncated kernel, odd numbers of supporting points are preferable.

Another idea to make the sinc function usable for spatial convolution might be to use it with a less severe window $w(x)$ than the rect function. Ostuni *et al.* discuss the use of a cosine function $w(x) = \cos(\pi x/N)$ for reslicing fMRI data [26]; Schaum uses a Hanning window, which is just a raised cosine, to taper the interpolation kernel's edges and remove Gibbs's overshoot in the transform [12]; and Wolberg compares several window functions for interpolation with windowed sinc kernels by Fourier analysis [13]. A systematic approach on the use of windows for harmonic analysis with the discrete Fourier transform is given by Harris who declared the Kaiser–Bessel and Blackman–Harris windows to be the top performers [27]. When using the three-term Blackman–Harris window

$$w(x) = w_0 + w_1 \cos\left(2\pi \frac{2x}{N}\right) + w_2 \cos\left(2\pi \frac{4x}{N}\right) \quad (10)$$

with $N = 6$ and

$$\begin{aligned} w_0 &= 0.423\,23 \\ w_1 &= 0.497\,55 \\ w_2 &= 0.079\,22 \end{aligned}$$

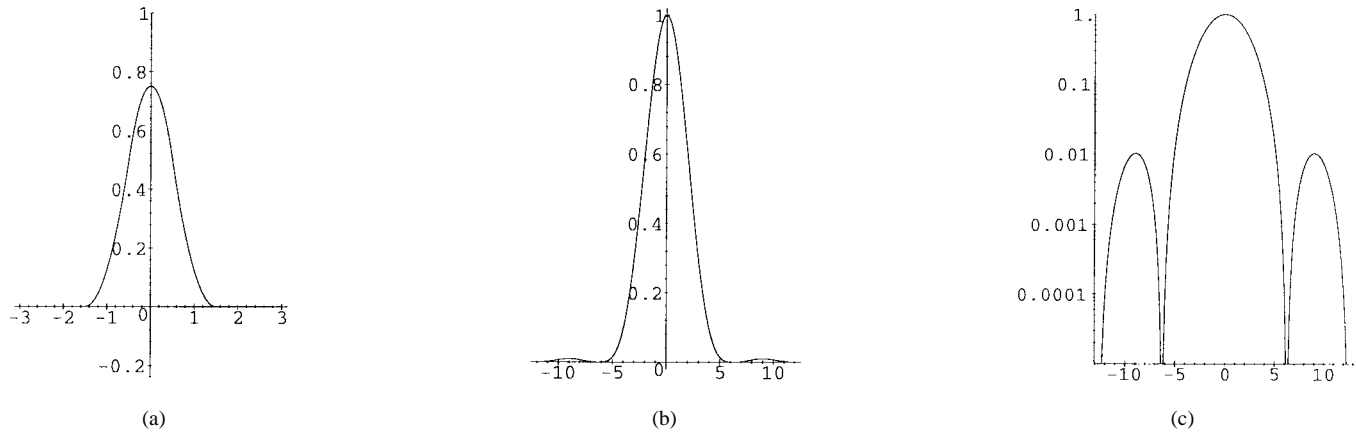


Fig. 11. Quadratic approximation, $a = 1/2$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

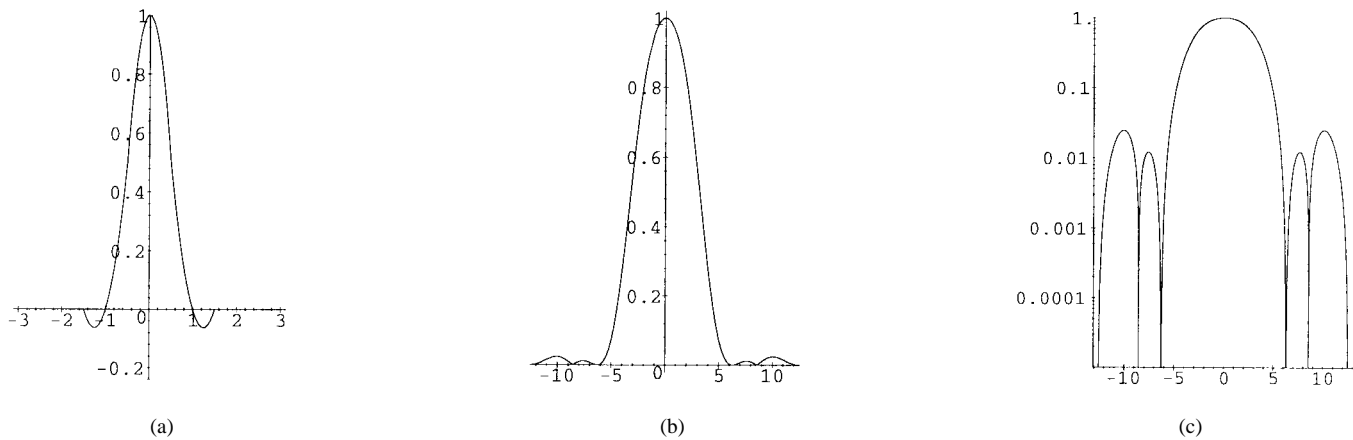


Fig. 12. Quadratic interpolation, $a = 1$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

a DC-constant interpolator is obtained. Note that most other window functions, including those used in [12] and [13] result in kernels that do not have this superior property.

Fig. 8 shows the Blackman–Harris windowed sinc kernel. The kernel's half wave between $2 \leq |x| < 3$ is suppressed significantly in comparison with the ideal or truncated kernels. Therefore, the ripples in the stopband are below 0.01%, but higher frequencies within the passband are attenuated also. The largest gain within the stopband is 0.5 at the cutoff point.

C. Nearest Neighbor Interpolation

The easiest way to approximate the sinc function by a spatially limited kernel is given by the nearest neighbor method. The value $s(x)$ at the location (x) is chosen as the next known value $s(k)$. Therefore, only $N = 1$ supporting point is required for the nearest neighbor interpolation. This is tantamount to convolution with a rect function [Fig. 9(a)]

$$h_1(x) = \begin{cases} 1, & 0 \leq |x| < 0.5 \\ 0, & \text{elsewhere.} \end{cases} \quad (11)$$

Clearly, $h_1(x)$ is a DC-constant interpolator.

Fig. 9(b) shows that the Fourier spectrum of the nearest neighbor kernel equals the sinc function (expressed in the frequency domain). The logarithmical scale shows prominent

sidelobes in those regions of the frequency domain where the repetitions of S caused by scanning s should be suppressed [Fig. 9(c)]. The gain in the passband rapidly falls off to $2/\pi \approx 64\%$ at the cutoff point, and the amplitude of the side maxima is more than 20%. Therefore, strong aliasing and blurring effects are associated with the nearest neighbor method for image interpolation.

D. Linear Interpolation

For separated bi-linear interpolation, the values of both direct neighbors are weighted by their distance to the opposite point of interpolation. Therefore, the linear approximation of the sinc function follows the triangular function

$$h_2(x) = \begin{cases} 1 - |x|, & 0 \leq |x| < 1 \\ 0, & \text{elsewhere.} \end{cases} \quad (12)$$

The triangular function $h_2(x)$ corresponds to a modest low-pass filter $H_2(f)$ in the frequency domain (Fig. 10). Again, $h_2(0) = 1$, $h_2(\pm 1, 2, \dots) = 0$, $H_2(0) = 1$, and $H_2(\pm 1, 2, \dots) = 0$. Therefore, the linear kernel is a DC-constant interpolator. The sidelobes in the stopband are below 10%, which is still considerable. Therefore, the main disadvantages of linear interpolation are both the attenuation of

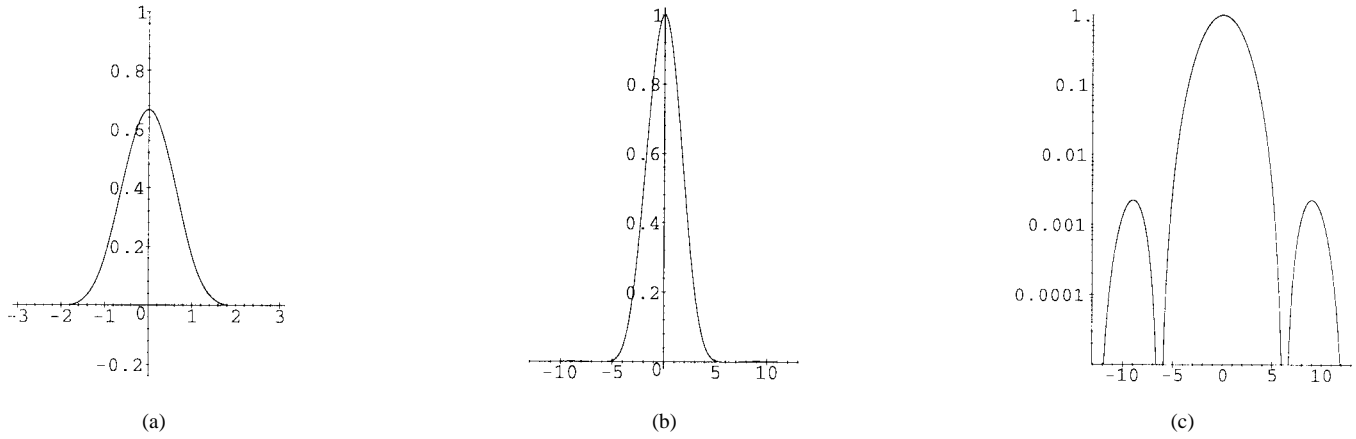


Fig. 13. Cubic B-spline approximation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

high-frequency components and the aliasing of the data beyond the cutoff point into the low frequencies [7].

E. Quadratic Approximation

One of the most frequently applied concepts to create sinc like interpolation kernels is the use of algebraic polynomials. Their advantage is easy determination and uniform approximation of continuous functions at finite intervals. In the previous sections, constant and linear polynomials have been discussed. Quadratic functions have been disregarded largely because they have been thought to introduce phase distortions. In fact, if the polynomials span -1 to 2 , asymmetric kernels with nonlinear phases are produced [28]. However, Dodgson showed this not to be the general case and recently derived a family of quadratic functions that is better behaved [11]. In contrast with other polynomial interpolation methods, this quadratic family is based on a symmetric 3×3 neighborhood from $-3/2$ to $3/2$, and the contacting points are fractions. In terms of separated kernels, the three nearest points are used for interpolation. Both direct neighbors and a third point sometimes are located on the left-hand side and other times on the right.

A symmetric quadratic kernel is given by

$$\text{Quad}h_3(x) = \begin{cases} +A_1|x|^2 & +B_1|x| & +C_1, & 0 \leq |x| < 1/2 \\ +A_2|x|^2 & +B_2|x| & +C_2, & 1/2 \leq |x| < 3/2 \\ 0, & \text{elsewhere} \end{cases}$$

with $A_i, B_i, C_i \in \mathbb{R}$ and $B_1 = 0$ due to C1-continuity (we call a function Cx continuous if its x th derivation is holomorph). To form a kernel useful for interpolation, additional restrictions must be imposed. The polynomials should fit exactly at the kernel's starting and ending points as well as at their contacting points. In addition, (6) must be satisfied to obtain

a DC-constant kernel. Hence, the following four equations are required to establish appropriate values for the five remaining parameters:

$$\begin{aligned} \text{Quad}h_3\left(\frac{1^-}{2}\right) &= \text{Quad}h_3\left(\frac{1^+}{2}\right) \Rightarrow \frac{A_1}{4} + \frac{B_1}{2} + C_1 \\ &= \frac{A_2}{4} + \frac{B_2}{2} + C_2 \\ \text{Quad}h_3\left(\frac{3}{2}\right) &= 0 \Rightarrow \frac{9A_2}{4} + \frac{3B_2}{2} + C_2 = 0 \\ \sum_k \text{Quad}h_3(d+k) &\equiv 1 \Rightarrow \begin{cases} 2A_2 + A_1 = 0 \\ 2A_2 + 2B_2 + 2C_2 + C_1 = 1 \end{cases} \end{aligned}$$

which reduce the general quadratic form to one degree of freedom, $a \in \mathbb{R}$ [see (13) at the bottom of this page].

Note that all members of this one parameter family of quadratics are real and even in the spatial domain and therefore have a linear phase in the frequency domain. To remove the final degree of freedom we can force the first derivatives of the polynomials to fit at their contact points $|x| = 1/2$. This sets $a = 1/2$. The resulting kernel (Fig. 11) does not satisfy condition (5). For that, the $a = 1/2$ quadratic kernel is called a quadratic approximator. By definition, the quadratic approximator is DC-constant. It has a prominent sidelobe in the Fourier domain with an amplitude of about 1%. The main lobe still considerably deviates from the ideal rectangular shape. Higher frequencies within the passband are attenuated. Therefore, interpolation with the quadratic approximator will cause strong blurring effects.

F. Quadratic Interpolation

One also can use the single degree of freedom in (13) to force the quadratic kernel to satisfy the zero-crossing condition (5). In this case, $a = 1$ is obtained and the finite kernel is

$$\text{Quad}h_3(x) = \begin{cases} -2a|x|^2 & +1/2(a+1), & 0 \leq |x| < 1/2 \\ +a|x|^2 & -(2a+1/2)|x| & +3/4(a+1), & 1/2 \leq |x| < 3/2 \\ 0, & \text{elsewhere} \end{cases} \quad (13)$$

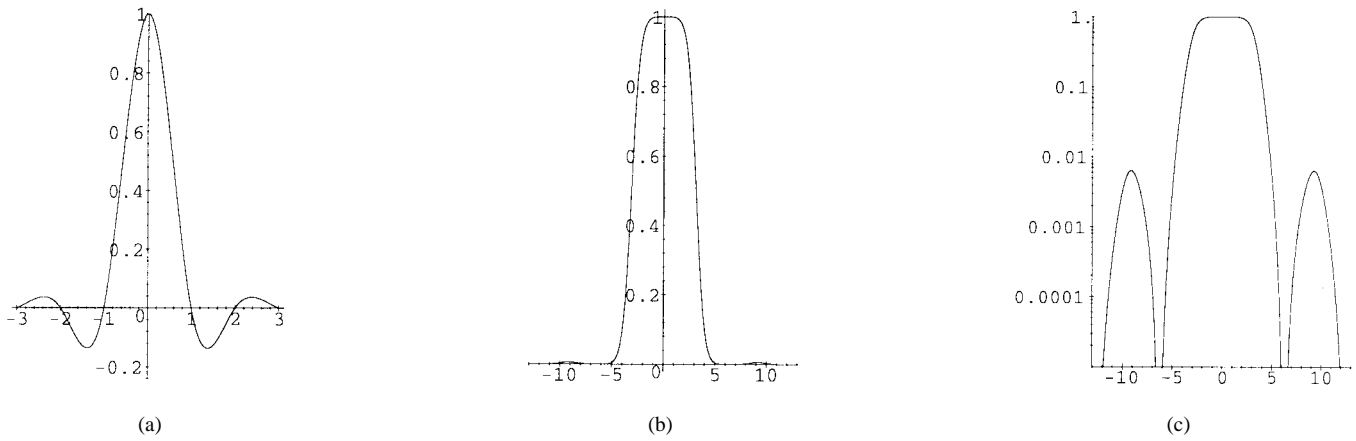


Fig. 14. Cubic B-spline interpolation. (a) Kernel plotted for $|x| < 3$. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

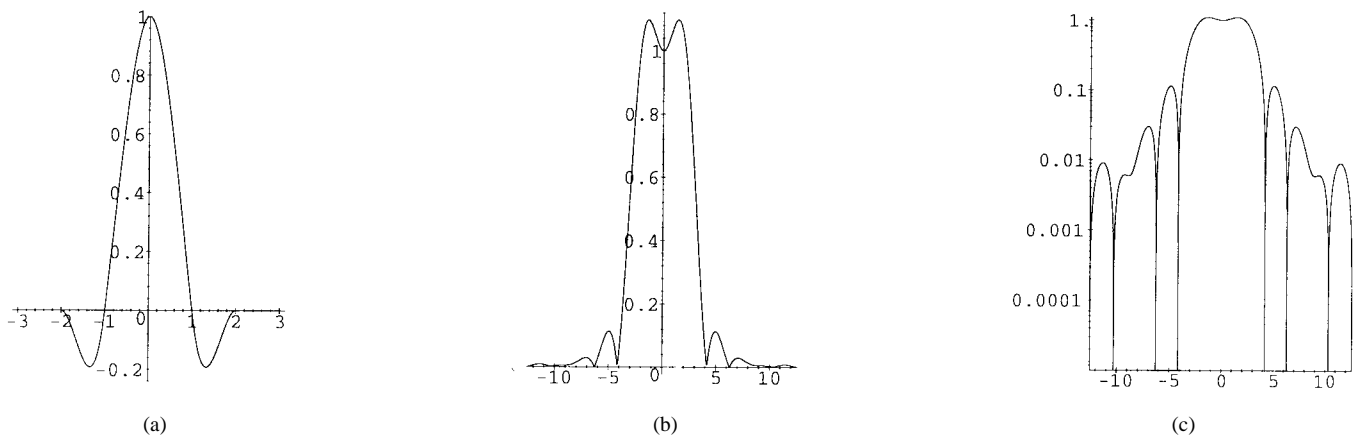


Fig. 15. Cubic interpolation, $N = 4$, $a = -1.3$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

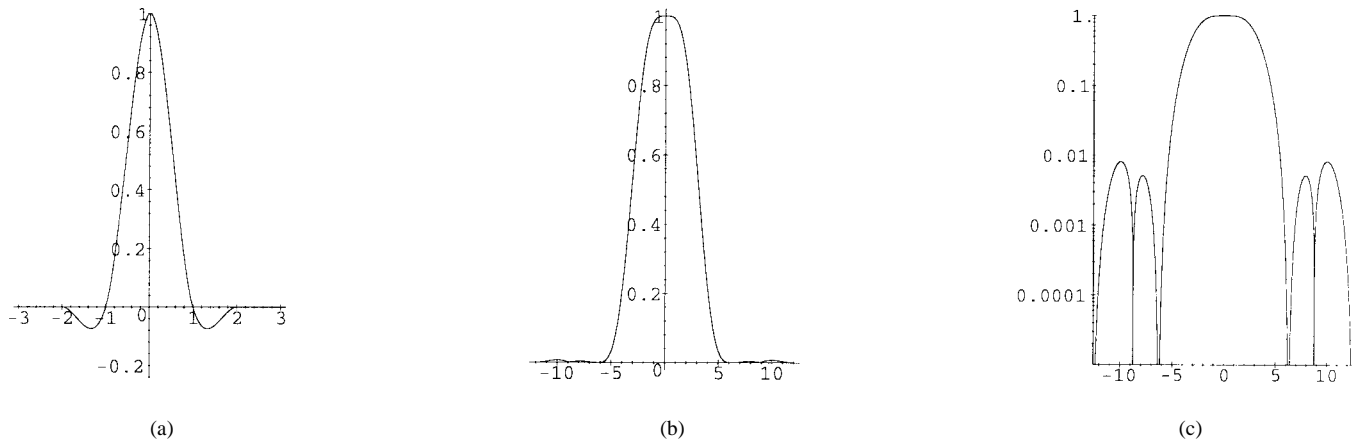


Fig. 16. Cubic interpolation, $N = 4$, $a = -1/2$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

a DC-constant interpolator. The kernel has a little nook at $|x| = 1/2$. However, the nook is covert in Fig. 12(a) because of the scaling of the plot. Note, that condition (5) can not be enforced at the same time as the C1-continuity. This C1-discontinuity raises the kernel's stopband attenuation for high frequencies above 2% [Fig. 12(c)]. However, the quadratic interpolator shows acceptable passband properties [Fig. 12(b)].

G. B-Spline Approximation

Basis splines (B-splines) are one of the most commonly used family of spline functions [4]. They can be derived by several self-convolutions of a so called basis function. Actually, the linear interpolation kernel $h_2(x)$ from (12) can be considered as the result of convolving the rectangular nearest neighbor

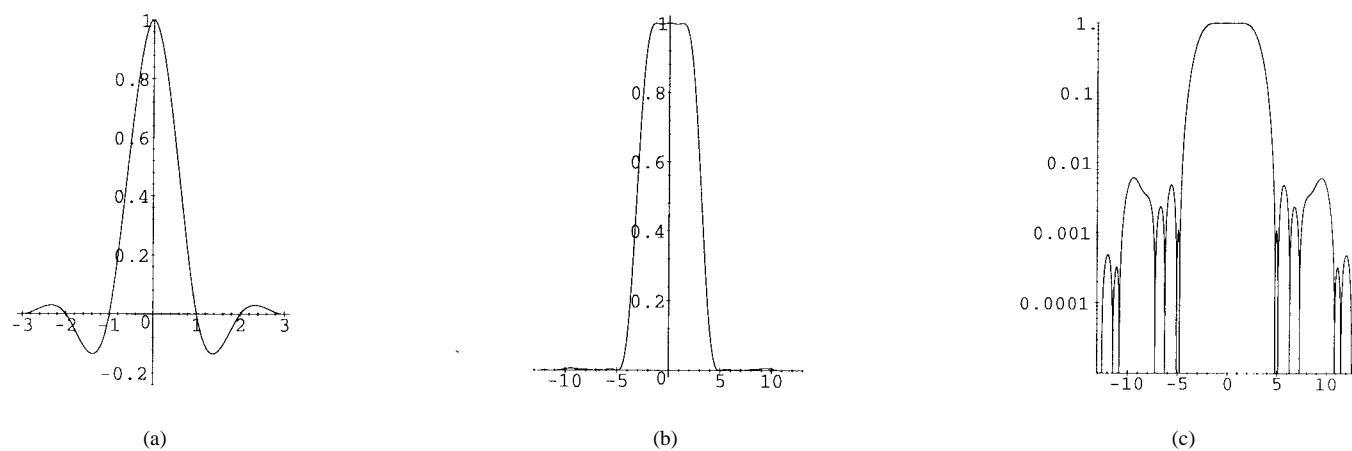


Fig. 17. Cubic interpolation, $N = 6$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

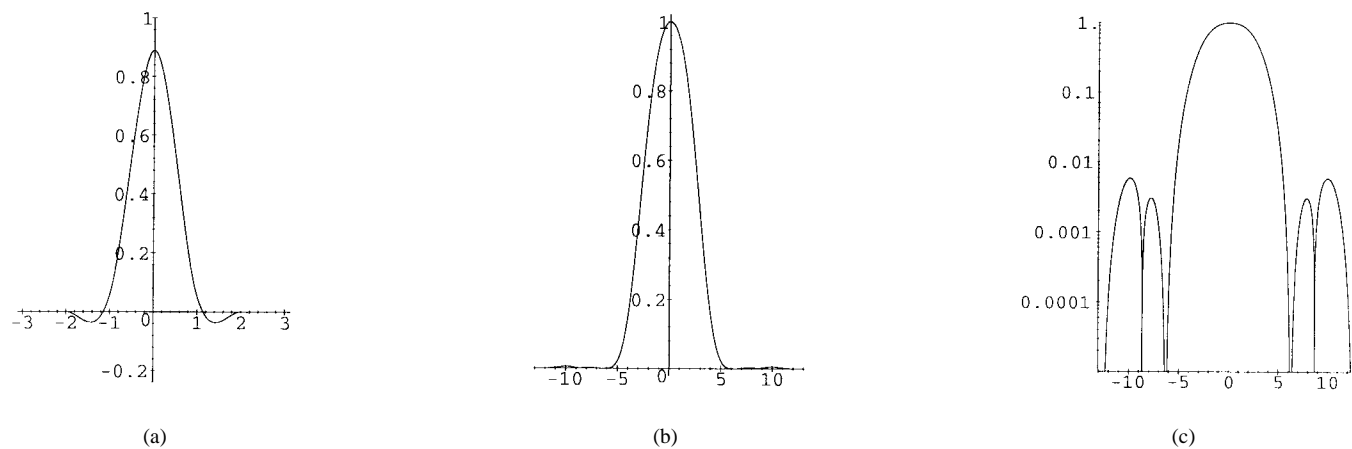


Fig. 18. Mitchell and Netravali's subjective best interpolation, $b = 1/3$, $c = 1/3$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

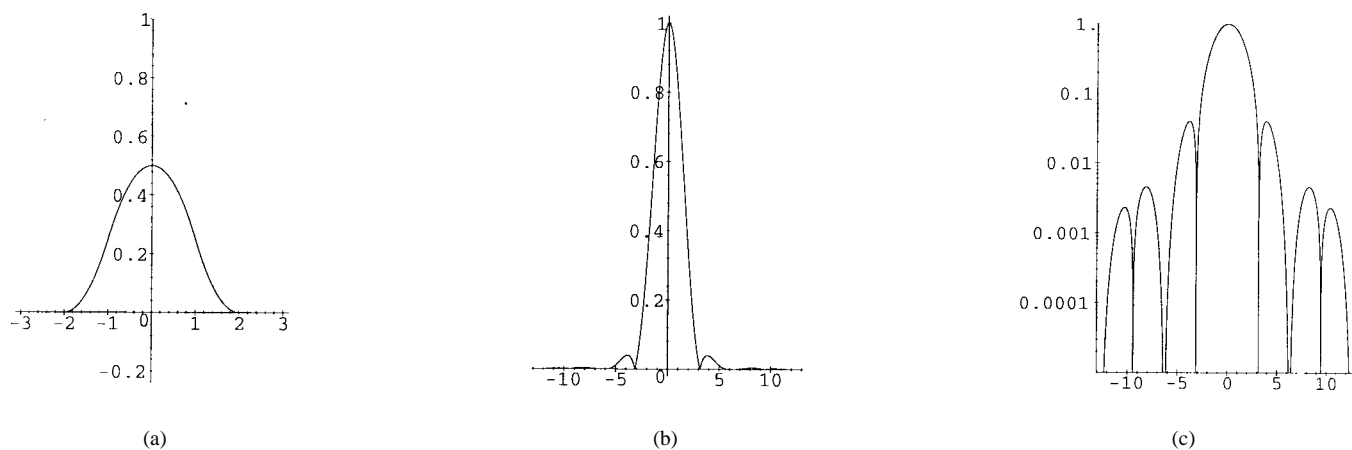


Fig. 19. Notch filter, $b = 3/2$, $c = -1/4$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

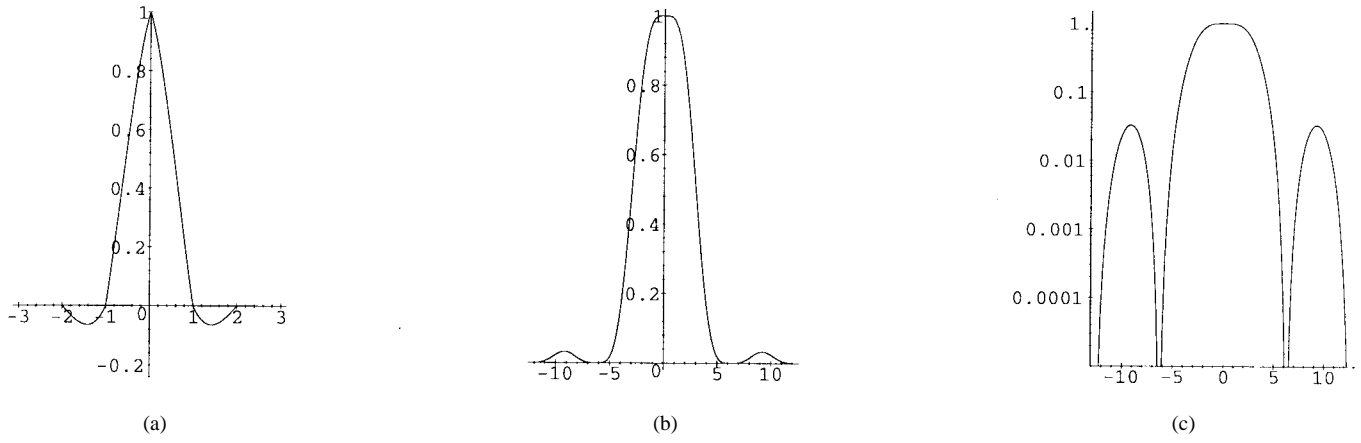


Fig. 20. Lagrange third-order interpolation, $N = 4$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

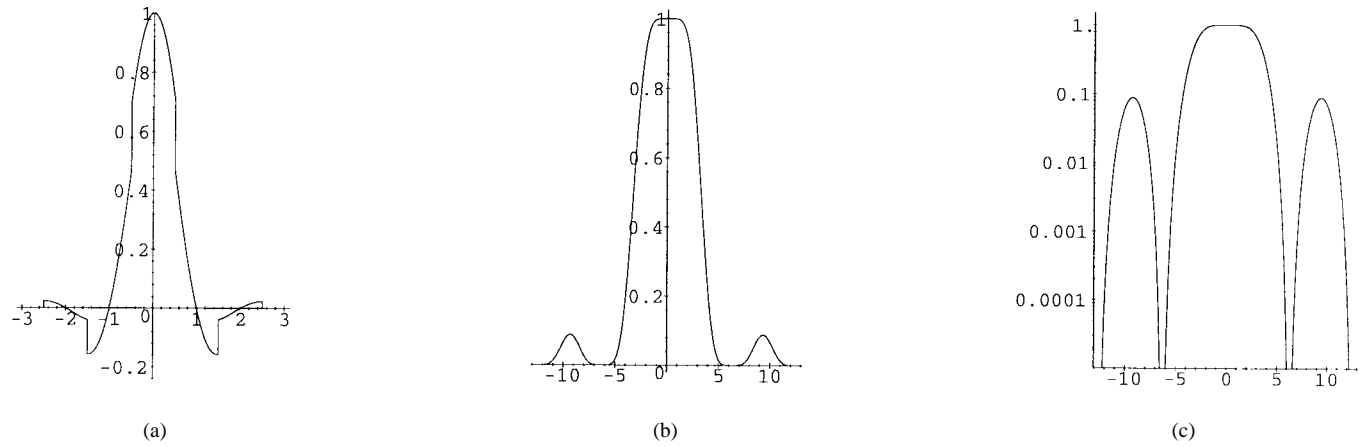


Fig. 21. Lagrange fourth-order interpolation, $N = 5$. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

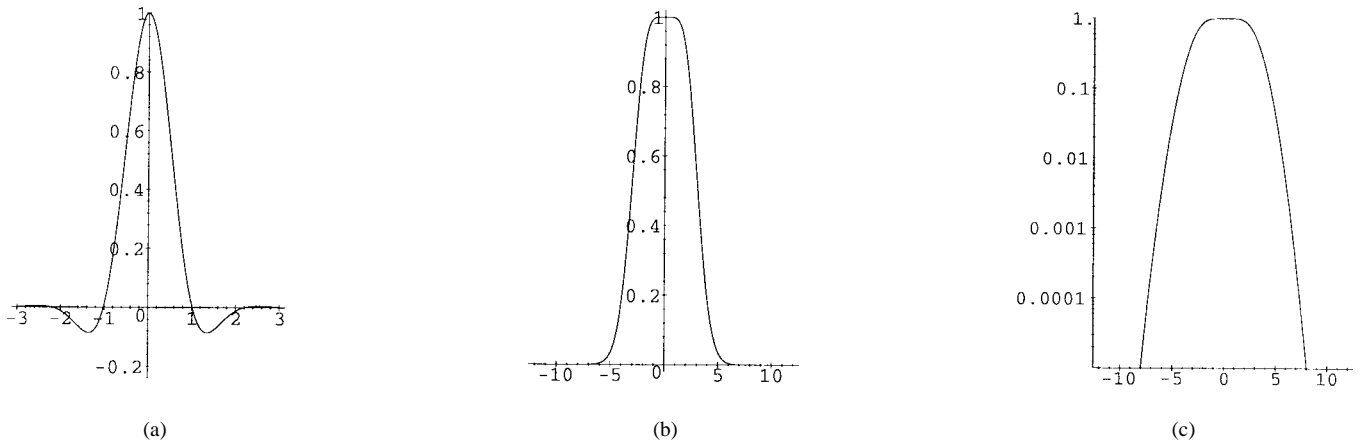


Fig. 22. Gaussian second-order interpolation. (a) Kernel plotted for $|x| < 3$. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

kernel $h_1(x)$ from (11) with itself

$$h_2(x) = h_1(x) * h_1(x)$$

Therefore, the rect function $h_1(x)$ can be used for the construction of uniform B-splines $h_N(x)$ of order N

$$h_N(x) = \underbrace{h_1(x) * h_1(x) * \dots * h_1(x)}_{N-1 \text{ times}}$$

Subsequently, for $N \rightarrow \infty$ this process converges to a Gaussian function $h_\infty(x)$. For $N = 3$ we obtain the quadratic B-spline $h_3(x)$ which, in fact, equals the previously mentioned quadratic approximator $^{\text{Quad}}h_3(x)$ for $a = 1/2$. For $N = 4$ we obtain the cubic B-spline [4] [see (14)].

$$h_4(x) = \begin{cases} (1/2)|x|^3 & -|x|^2 & +2/3, & 0 \leq |x| < 1 \\ -(1/6)|x|^3 & +|x|^2 & -2|x| & +4/3, & 1 \leq |x| \leq 2 \\ 0, & \text{elsewhere.} \end{cases} \quad (14)$$

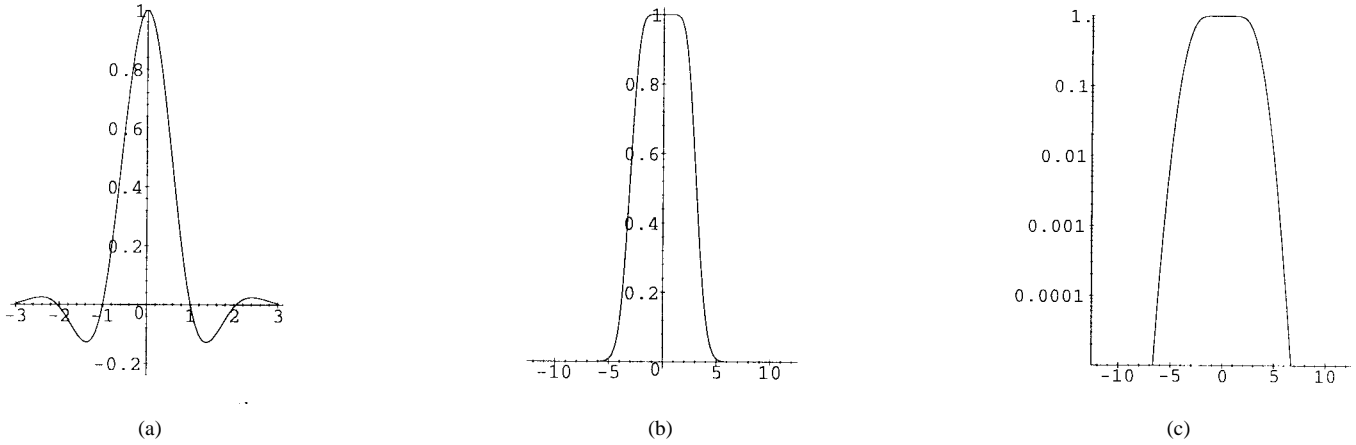
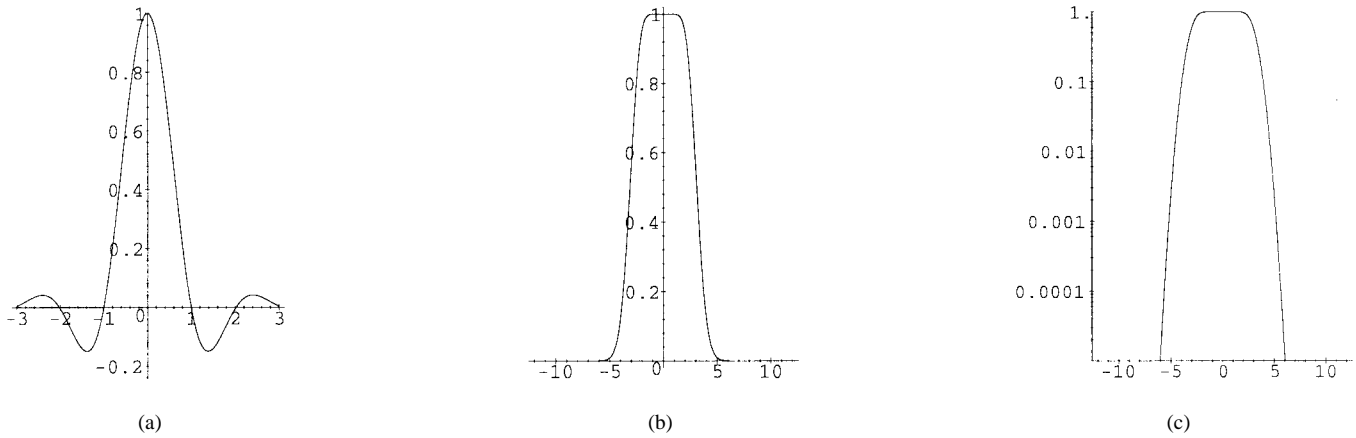


Fig. 23. Gaussian sixth-order interpolation. (a) Kernel. (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.


 Fig. 24. Gaussian tenth-order interpolation. (a) Kernel plotted for $|x| < 3$, (b) Magnitude of Fourier transform. (c) Logarithmic plot of magnitude.

Note that the B-spline kernel $h_4(x)$ fails to occupy the null positions of the sinc function: $h_4(-1) = h_4(1) = 1/6$ and, therefore, the B-spline kernel is actually not an interpolation but rather an approximation kernel. Consequently, B-spline approximation displays strong blurring effects but also allows for the attenuation of unwanted high-frequency noise in the output image [29].

However, the B-spline approximator enjoys the valuable capacity to retain the properties of DC-amplification [Fig. 13(c)]. Furthermore, Fig. 13 demonstrates that the cubic B-spline function has a favorable stopband response. The amplitude of the sidelobes is lower than 1%. Nevertheless, the Fourier transform of the cubic B-spline kernel is equivalent to a $(\text{sinc})^4$ function that results in over smoothing in the passband. The gain at the cutoff frequency is only about 16.4%. Increasing the order of the spline not only improves the quality of interpolation but also increases the smoothing effects. Therefore, h_4 was selected for this study to represent the B-spline approximation method. This choice corresponds to those of other authors [4], [5], [7], [11].

H. B-Spline Interpolation

To create an interpolating B-spline kernel, the B-spline approximator is applied to a different set of samples $t(k)$. Since the B-spline kernel is symmetrical and separable, the

reconstruction (2) yields

$$s(x) = \sum_k t(k) \cdot h(x - k) \quad (15)$$

with $h = h_4$, as defined in (14). Note that the general case (15) reduces to (2) if the samples are taken directly from the image data: $t(k) \equiv s(k)$.

Here, the $t(k)$ must be derived from the image's sample points $s(k)$ in such a way that the resulting curve interpolates the discrete image. From (15) and (14) we obtain

$$\begin{aligned} s(k) &= \sum_{m=k-2}^{k+2} t(m) \cdot h_4(k - m) \\ &= \frac{1}{6}(t(k-1) + 4t(k) + t(k+1)) \end{aligned} \quad (16)$$

which, ignoring edge effects, results in a set of equations to solve

$$\begin{bmatrix} s(0) \\ s(1) \\ s(2) \\ \vdots \\ s(K-2) \\ s(K-1) \end{bmatrix} = \frac{1}{6} \begin{bmatrix} 4 & 1 & & & 0 \\ 1 & 4 & 1 & & \\ & 1 & 4 & 1 & \\ & & \ddots & \ddots & \\ & & & 1 & 4 & 1 \\ 0 & & & & 1 & 4 \end{bmatrix} \cdot \begin{bmatrix} t(0) \\ t(1) \\ t(2) \\ \vdots \\ t(K-2) \\ t(K-1) \end{bmatrix}. \quad (17)$$

Labeling the three matrices above as \underline{S} , \underline{C} , and \underline{T} , respectively, the coefficients in \underline{T} may be evaluated by multiplying the known data points \underline{S} with the inverse of the tridiagonal matrix \underline{C}

$$\underline{T} = \underline{C}^{-1} \cdot \underline{S}. \quad (18)$$

In all other methods included in this paper, the coefficients used for convolution with the interpolation kernel are taken to be the data samples themselves. Because the coefficients for B-spline interpolation are determined by solving a tridiagonal matrix system, the resulting kernel $\text{Spline}h(x)$ is infinite.

To simplify its analytical derivation, the interpolated image $s(x)$ and the data samples $s(k)$ now are called u and v , respectively. From (16) we obtain $v = t * c$ and in the frequency domain

$$\begin{aligned} v(x) &= t(x) * \frac{1}{6}(\delta(x-1) + 4\delta(x) + \delta(x+1)) \\ V(f) &= T(f) \cdot \frac{1}{6}(4 + 2\cos(2\pi f)). \end{aligned} \quad (19)$$

Inversion of (19) yields

$$\begin{aligned} T(f) &= V(f) \cdot \frac{3}{2 + \cos(2\pi f)} \\ &= V(f) \cdot \sqrt{3} \left(1 + \sum_{m=1}^{\infty} 2(\sqrt{3} - 2)^m \cos(2\pi m f) \right) \\ t(x) &= v(x) * \left(\sqrt{3}\delta(x) + \sum_{m=1}^{\infty} \sqrt{3}(\sqrt{3} - 2)^m \right. \\ &\quad \left. \cdot (\delta(x-m) + \delta(x+m)) \right). \end{aligned} \quad (20)$$

Hence, (15) can be written as $u = t * h_4 = v * \text{Spline}h$ and with (20) we finally obtain

$$\text{Spline}h(x) = h_4(x) * \sum_{m=-\infty}^{\infty} \sqrt{3}(\sqrt{3} - 2)^{|m|} \delta(x+m). \quad (21)$$

Fig. 14(a) plots $\text{Spline}h(x)$ within the interval $|x| < 3$. Although the kernel is infinite, the amplitudes of the half waves are reduced significantly when compared with that of the ideal IIR-interpolator [Fig. 5(a)]. The cubic B-spline interpolation shows excellent passband characteristics [Fig. 14(b)] and the amplitude of the sidelobes in the stopband is below 1% [Fig. 14(c)]. Note that $\text{Spline}h(x)$ satisfies the zero crossing condition (5) as well as the partition of unity condition (6) and, therefore, $\text{Spline}h(x)$ is a true DC-constant interpolator.

I. Cubic Interpolation

Cubic polynomials are used frequently because of their ability to fit C2-continuous. Also, the B-spline approximator h_4 , as defined in (14), as well as the Lagrange interpolator $\text{Lagra}h_4$ in (28), are constructed piecewise from cubic polynomials. Of course, cubic polynomials also can be used to approximate the sinc function.

1) *Two-Point Interpolation:* In the case of cubic interpolation with two points, a symmetric kernel can be defined with $A, B, C, D \in \mathbb{R}$

$$\text{Cubic}h_2(x) = \begin{cases} A|x|^3 + B|x|^2 + C|x| + D, & 0 \leq |x| < 1 \\ 0, & \text{elsewhere.} \end{cases}$$

The parameters A to D can be determined by applying the following boundary conditions:

- $h(k^-) = h(k^+)$, C0-continuity;
- $h'(k^-) = h'(k^+)$, C1-continuity;
- $h(k) = 1$ for $k = 0$, see (5);
- $h(k) = 0$ for $k \neq 0$, see (5).

For $N = 2$, those boundary conditions yield four equations for the four parameters resulting in ;

$$\text{Cubic}h_2(x) = \begin{cases} 2|x|^3 - 3|x|^2 + 1, & 0 \leq |x| < 1 \\ 0, & \text{elsewhere.} \end{cases} \quad (22)$$

It should be pointed out that, by definition, $\text{Cubic}h_2(x)$ is a DC-constant interpolator. The resulting curves are similar to those obtained by linear interpolation, but the pieces fit C1-continuously in the spatial domain. Two sidelobes occur in the positive Fourier plane. The first lobe raises up to 2%. It is located within $1 \leq |f| < 1.5$ while the second's amplitude is about 1% between $1.5 \leq |f| < 2$.

2) *Four-Point Interpolation:* When using cubic interpolation with $N = 4$ points, seven of the eight coefficients are determined from the above constraints and one extra free parameter a is retained [see (23) at the bottom of the next page].

Different concepts have been used to determine this parameter. With the constant negative, the kernel is positive in the interval from zero to one and negative from one to two and, hence, an interpolator is obtained.

According to Danielsson and Hammerin, when $a = -1.3$, the result is a kernel $\text{Cubic}h_4(x)$ whose Fourier transform deviates minimally from the ideal rectangular function [8], [9]. Fig. 15 shows the corresponding interpolation function in the spatial and Fourier domains. Frequencies directly below the cutoff point are amplified slightly, and the transition between the passband and the stopband is quite sharp. The amplitude of the first sidelobe is above 10%, but the ripples reduce below 1% for $f > 1.3$.

If a is determined in order to match the slope of the sinc function at $x = 1$, then $a = -1$ is obtained [29], [30]. The intensification of frequencies just below the cutoff point is reduced when compared to the situation that $a = -1.3$ is chosen, but the transition between the passband and the stopband is not as sharp as before. This trend continues if a is increased further. If the second derivatives $\text{Cubic}h_4''(x)$ of both polynomials in (23), shown at the bottom of the page, are made equal for their contact point of $x = 1$, then $a = -3/4$ is obtained. Park and Schowengerdt found $a = -2/3$ to minimize the sampling and reconstruction error for images dominated by edges [30].

Keys determined the constant a by forcing the Taylor series expansion of the sampled sinc function to agree in as many terms as possible with the original signal resulting in $a = -1/2$ [5]. When using this choice of a , the first three terms of

the Taylor series expansion of the input signal agree with the interpolated function. Thus, cubic interpolation with $a = -1/2$ can reconstruct any second-degree polynomial. Furthermore, the approximation error is proportional to the third power of the sampling increment [5]. Only in the case of $a = -1/2$ does the $N = 4$ cubic kernel not have any overshoots within the passband. Fig. 16 shows the flat spectrum at low frequencies, which fall off to the cutoff frequency. Only two sidelobes appear in the positive Fourier half plane, and the amplitude of both distinct lobes is below 1%. Because for most digital images a preponderance of energy exists at low frequencies, Park and Schowengerdt also derived $a = -1/2$ to be optimal for the image-independent case [30]. Reichenbach and Park, as well as Dodgson, showed that this choice for the parameter a corresponds to both the Catmull–Rom blended spline and the piecewise cubic Hermite interpolation [31], [32]. In addition, Dodgson pointed out that the Bezier form of the cubic spline interpolation is related to $a = -3/8$ [32].

3) *Six-Point and Eight-Point Interpolation*: Increasing the interpolation kernel size N improves the quality of resampling. From the boundary conditions defined above, only $3N/2 + 1$ equations are obtained to determine the $2N$ parameters. Hence, there are two degrees of freedom for $N = 6$ and three free parameters for $N = 8$. Again, many ideas have been used to determine these parameters. Danielsson and Hammerin forced the kernel to have its first minimum value at the same position as the sinc function [8], [9]. With the kernel suggested by Keys, the sidelobes are further subdued, and the transition between the passband and the stopband is sharpened when compared to the 4×4 kernels [5].

Nevertheless, not all of those concepts are generic for every choice of N . In addition, it is doubtful whether any *a priori* attempt to fit the kernel to the sinc function results in a valuable interpolation scheme. This is true because the implicit supposition, i.e., the applicability of the sampling theorem, is violated often in medical imaging systems. However, the missing smoothness of the kernel is responsible for unwanted ripples in the Fourier domain. Exactly $N/2 - 1$ independent equations are obtained if the second derivatives of the poly-

nomials are forced to match for all contacting points. Using those boundary conditions, we obtain (24) and (25), shown at the bottom of this page, for $N = 6$ and $N = 8$, respectively. Fig. 17 shows the cubic interpolation kernel corresponding to (24). Compared with the other examples of cubic kernels, the plateau of the passband is enlarged and sharp edged while the amplitudes of the sidelobes are further reduced. Higher frequencies within the passband are amplified somewhat.

The reader should notice that only DC-constant interpolators have been derived in this subsection. Equation (5) is fulfilled by the definition of the boundary conditions and the partition of unity condition (8) is satisfied for all kernel sizes and parameters (Figs. 15–17).

J. Mitchell and Netravali's Method

Mitchell and Netravali developed a family of $N = 4$ point cubic filters that can be either approximators or interpolators [33]. The constraints they use are the following:

- $h(k^-) = h(k^+)$, CO-continuity;
- $h'(k^-) = h'(k^+)$, C1-continuity;
- $\sum_k h(d+k) = 1$, see partition of unity condition (6).

That leaves us with a two-parameter family of solutions [13], [33] [see (26) at the bottom of this page].

Several well known cubic filters are derivable from (26) through an appropriate choice of the parameter tuple (b, c) . For instance, $(1, 0)$ is the cubic B-spline approximator (14), and $(0, -a)$ corresponds to the four-point cubic interpolator family (23).

Mitchell and Netravali partitioned the parameter space into regions characterizing artefacts, such as blurring, anisotropy, and ringing. Measured by subjective inspection, the tuple $(1/3, 1/3)$ was found to offer superior image quality [33]. Fig. 18 shows Mitchell and Netravali's subjective best kernel, which is not an interpolator. However, it is a DC-constant kernel. Higher frequencies within the passband are absorbed and the image is smoothed during resampling. The two sidelobes in the stopband are similar to those of the cubic interpolator shown in Fig. 16.

$$\text{Cubic } h_4(x) = \begin{cases} (a+2)|x|^3 & -(a+3)|x|^2 & +1, & 0 \leq |x| < 1 \\ a|x|^3 & -5a|x|^2 & +8a|x| & -4a, & 1 \leq |x| < 2 \\ 0, & \text{elsewhere} \end{cases} \quad (23)$$

$$\text{Cubic } h_6(x) = \begin{cases} (6/5)|x|^3 & -(11/5)|x|^2 & +1, & 0 \leq |x| < 1 \\ -(3/5)|x|^3 & +(16/5)|x|^2 & -(27/5)|x| & +14/5, & 1 \leq |x| < 2 \\ (1/5)|x|^3 & -(8/5)|x|^2 & +(21/5)|x| & -18/5, & 2 \leq |x| < 3 \\ 0, & \text{elsewhere} \end{cases} \quad (24)$$

$$\text{Cubic } h_8(x) = \begin{cases} (67/56)|x|^3 & -(123/56)|x|^2 & +1, & 0 \leq |x| < 1 \\ -(33/56)|x|^3 & +(177/56)|x|^2 & -(75/14)|x| & +39/14, & 1 \leq |x| < 2 \\ (9/56)|x|^3 & -(75/56)|x|^2 & +(51/14)|x| & -45/14, & 2 \leq |x| < 3 \\ -(3/56)|x|^3 & +(33/56)|x|^2 & -(15/7)|x| & +18/7, & 3 \leq |x| < 4 \\ 0, & \text{elsewhere} \end{cases} \quad (25)$$

$$\text{Mitch } h_4(x) = \frac{1}{6} \cdot \begin{cases} -(9b+6c-12)|x|^3 & +(12b+6c-18)|x|^2 & -(2b-6), & 0 \leq |x| < 1 \\ -(b+6c)|x|^3 & +(6b+30c)|x|^2 & -(12b+48c)|x| & +(8b+24c), & 1 \leq |x| < 2 \\ 0, & \text{elsewhere} \end{cases} \quad (26)$$

Another parameter tuple suggested by Mitchell and Netravali is $(3/2, -1/4)$ yielding the so-called notch filter (Fig. 19). The frequency response of this DC-constant approximator is zero at all integer and half-integer multiples of the sampling rate except zero. On the one hand, the notch kernel cuts signal energy very near to the Nyquist frequency $f = 1/2$ or $\omega = \pi$, which contributes primarily to aliasing effects and Moiré patterns [33]. On the other, the gain within the passband drops off quickly resulting in strongly blurred images.

K. Lagrange Interpolation

In numerical analysis, the Taylor polynomials are used frequently. As a major drawback for image resampling, the Taylor polynomials are expanded from a single point, which might cause problems when they are applied to image interpolation. Using Lagrange polynomials instead, several points through which the polynomial must pass can be specified. Particularly, the Lagrange polynomial of degree $N - 1$ passes through N points [3]. For an infinite number of points these polynomials approach the sinc function [2]

$$\text{sinc}(x) = \prod_{i=-\infty, i \neq 0}^{\infty} \left(1 + \frac{x}{i}\right) = \prod_{i=1}^{\infty} \left(1 - \frac{x^2}{i^2}\right)$$

Therefore, the Lagrange interpolation formula often is confounded with a terminated product representation of the sinc function giving bad interpolation results [1].

The Lagrange interpolation kernel presented here refers to various textbooks on numerical analysis. The Lagrange kernel of degree $N - 1$ for an $N \times N$ region with $n \in \{-N/2 + 1, -N/2 + 2, \dots, N/2\}$ is defined by

$$\text{Lagra}h_N(x) = \begin{cases} \prod_{j=0, j-\frac{N}{2}+1 \neq n}^{N-1} \frac{n-i-x}{n-i}, & n-1 \leq x < n \\ 0, & \text{elsewhere} \end{cases} \quad (27)$$

with $i = j - N/2 + 1$.

Because the neutral element of multiplications is one, the Lagrange kernel for $N = 1$ equals the nearest neighbor interpolation. One can easily show that in the case of $N = 2$ (27) equals the linear interpolation method. As mentioned above, the Lagrange kernel for $N = 4$ supporting points results in cubic polynomials [see (28) at the bottom of this page]. For $N = 5$, (27) yields fourth-order polynomials [see (29) at the bottom of this page].

A little algebra shows that all Lagrange kernels are DC-constant interpolators. However, the even Lagrange interpola-

tors do not fit C1-continuously at the connecting points $x = n$ causing significant sidelobes within their Fourier transforms. Figs. 20 and 21 visualize the Lagrange kernels for $N = 4$ and $N = 5$, respectively. $\text{Lagra}h_4(x)$ shows a sharp edge at the center of the mask in the spatial domain. The amplitude of the sidelobe in the Fourier domain is about 4%. The odd Lagrange kernel $\text{Lagra}h_5(x)$ is not CO-continuous. Therefore, the amplitude of the sidelobe is raised up to 10%. However, the plateau in the passband is wider causing the major lobe to approximate more closely the ideal rectangular shape. In other words, the passband characteristic is improved by raising the order of the Lagrange kernel. Furthermore, odd kernels should be used for scenes where high contrasts dominate.

L. Gaussian Interpolation

Appledorn has recently introduced a new approach to the generation of interpolation kernels [19]. The objective was to exploit the characteristics of the Gaussian function in both the spatial and the frequency domain. In particular, the Gaussian function is recurrent with respect to operations such as derivation and Fourier transform. Hence, Appledorn published a scheme to develop simple interpolation kernels that are both locally compact in the signal space and almost band limited in the frequency domain and, in addition, are easy to manipulate analytically.

Consequently, we will denote the M th partial derivative of the unit area Gaussian function

$$G^0(x, \beta) = \frac{1}{\sqrt{2\pi\beta}} e^{-x^2/2\beta}$$

with zero mean and variance β as

$$G^M(x, \beta) = \frac{\partial^M}{\partial x^M} G^0(x, \beta).$$

Hence we obtain

$$G^2(x, \beta) = \frac{1}{\beta^2} (x^2 - \beta) G^0(x, \beta)$$

$$G^6(x, \beta) = \frac{1}{\beta^6} (x^6 - 15\beta x^4 + 45\beta^2 x^2 - 15\beta^3) G^0(x, \beta)$$

$$G^{10}(x, \beta) = \frac{1}{\beta^{10}} (x^{10} - 45\beta x^8 + 630\beta^2 x^6 - 3150\beta^3 x^4 + 4725\beta^4 x^2 - 945\beta^5) G^0(x, \beta).$$

Then, the M th-order Gaussian interpolation kernel is given by

$$\text{Gauss}h_N^M(x) = \begin{cases} \sum_{m=0}^M \alpha_m \cdot G^m(x, \beta_m), & 0 \leq |x| < \frac{N}{2} \\ 0, & \text{elsewhere.} \end{cases} \quad (30)$$

$$\text{Lagra}h_4(x) = \begin{cases} (1/2)|x|^3 & -|x|^2 & -(1/2)|x| & +1, & 0 \leq |x| < 1 \\ -(1/6)|x|^3 & +|x|^2 & -(11/6)|x| & +1, & 1 \leq |x| < 2 \\ 0, & \text{elsewhere} \end{cases} \quad (28)$$

$$\text{Lagra}h_5(x) = \begin{cases} (1/4)|x|^4 & & -(5/4)|x|^2 & & +1, & 0 \leq |x| < 1/2 \\ -(1/6)|x|^4 & +(5/6)|x|^3 & -(5/6)|x|^2 & -(5/6)|x| & +1, & 1/2 \leq |x| < 3/2 \\ (1/24)|x|^4 & -(5/12)|x|^3 & +(35/24)|x|^2 & -(25/12)|x| & +1, & 3/2 \leq |x| < 5/2 \\ 0, & \text{elsewhere} \end{cases} \quad (29)$$

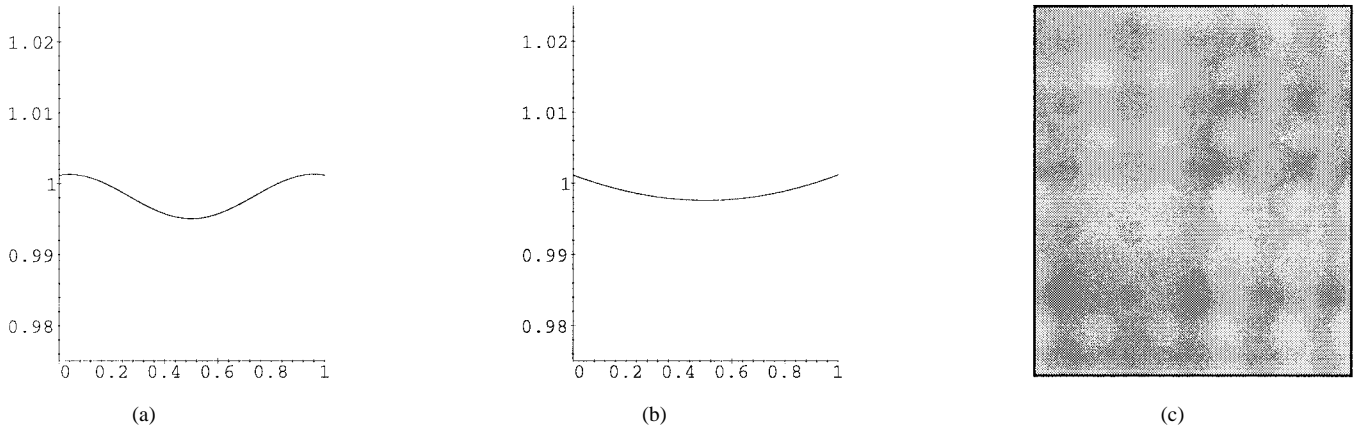


Fig. 25. Sum of sampled interpolation kernels as a function of the displacement d . (a) Gaussian second-order, $N = 6$. (b) Gaussian tenth-order, $N = 8$. (c) Result of scaling a constant image using Gaussian second-order interpolation (effects have been enhanced).

The weighting factors α_m and the variances β_m are determined with from the following constraints [19].

- The Gaussian kernels should equal the ideal interpolator, at least for $x = 0$

$$\text{Gauss } h_N^M(0) \equiv 1.$$

- The Fourier transforms $\text{Gauss } H_N^M(\omega)$ of the Gaussian kernels should equal those of the ideal interpolator, at least for $\omega = 0$

$$\text{Gauss } H_N^M(0) \equiv 1 \Leftrightarrow \int_{-\infty}^{\infty} \text{Gauss } h_N^M(x) dx \equiv 1.$$

- Furthermore, $\text{Gauss } H_N^M(\omega)$ should be as flat as possible without any slope or curvature for $\omega = 0$

$$\left. \frac{\partial^m}{\partial \omega^m} \text{Gauss } H_N^M(\omega) \right|_{\omega=0} = 0 \quad \text{for } m \neq 0 \text{ even.}$$

Note that the first and the second constraints cover only one part of the interpolation condition (5) and the DC-constant condition (8), respectively. The latter constraint is imposed to approximate the passband characteristics of the ideal low-pass filter and therefore to minimize the corruption of the image's Fourier spectrum by the interpolation. Hence, all weights α_m must be 0 if m is odd. In addition, $\alpha_m = 0$ for $m = 4, 8, 12, \dots$. Because the Gaussian function always approximates zero for large x , for the sake of simplicity, the explicit behavior of these values relative to the kernel's defined range in neglected. Accordingly, the first existing orders of Gaussian kernels are given by [19]

$$\text{Gauss } h_N^2(x) = G^0(x, 2\gamma_2) - \gamma_2 G^2(x, \gamma_2) \quad (31)$$

$$\text{Gauss } h_N^6(x) = G^0(x, 2\gamma_6) - \gamma_6 G^2(x, \gamma_6) - \frac{\gamma_6^3}{24} G^6(x, \gamma_6) \quad (32)$$

$$\text{Gauss } h_N^{10}(x) = G^0(x, 2\gamma_{10}) - \gamma_{10} G^2(x, \gamma_{10}) - \frac{\gamma_{10}^3}{24} G^6(x, \gamma_{10}) - \frac{\gamma_{10}^5}{1920} G^{10}(x, \gamma_{10}) \quad (33)$$

$$\begin{aligned} \gamma_2 &= \frac{1}{2\pi} \left(\frac{1}{\sqrt{2}} + 1 \right)^2 \approx 0.4638115 \\ \gamma_6 &= \frac{1}{2\pi} \left(\frac{1}{\sqrt{2}} + 1 + \frac{15}{24} \right)^2 \approx 0.8655995 \\ \gamma_{10} &= \frac{1}{2\pi} \left(\frac{1}{\sqrt{2}} + 1 + \frac{15}{24} + \frac{945}{1920} \right)^2 \approx 1.2695213. \end{aligned}$$

Although all kernels are one at $x = 0$, the zero crossings do not match exactly. Therefore, the Gaussian kernels are actually approximators. However, the deviation from the interpolator's general shape is quite small. Fig. 22 shows the infinite second-order Gaussian kernel $\text{Gauss } h^2(x)$ within the interval $-3 < x < 3$. The zero points are $x = 1.0186$ and $x = 2.1869$. Increasing the order M of the kernel improves the approximation of the ideal rectangular low-pass filter (Fig. 23). In other words, the passband is widened and the transition to the stopband is narrowed. Even for order $M = 2$, neither sidelobes nor ripples occur. The main lobe is flat for low frequencies but falls off with a broad slope. Fig. 24 displays the infinite tenth-order Gaussian kernel with nearly perfect frequency properties. Nevertheless, the first zero point of $\text{Gauss } h^{10}(x)$ is approximately at $x = 1.0026$ and $\text{Gauss } h^{10}(1) = 0.0021$ or $\text{Gauss } h^{10}(2) = -0.0049$.

Note that the sum of all sampled interpolation values is not equal to one for truncated Gaussian kernels and, hence, the Gaussian FIR-kernels actually are not DC-constant. Fig. 25(a) and (b) plots the sum of sample points from (6) as a function of the displacement d for $\text{Gauss } h_6^2(x)$ and $\text{Gauss } h_8^{10}(x)$, respectively. The distortion effect resulting from interpolation with DC-inconstant kernels is visualized in Fig. 25(c). Therefore, the Gaussian kernels have impressive frequency properties only when the approximation is created from enough points to reflect accurately the bulk of energy distribution (Figs. 22–24).

Fig. 26 shows the logarithmic plots of the Fourier magnitude for truncated Gaussian kernels. Because the ripples in the stopband are below 0.1%, effects resulting from truncation of Gaussian kernels are negligible.

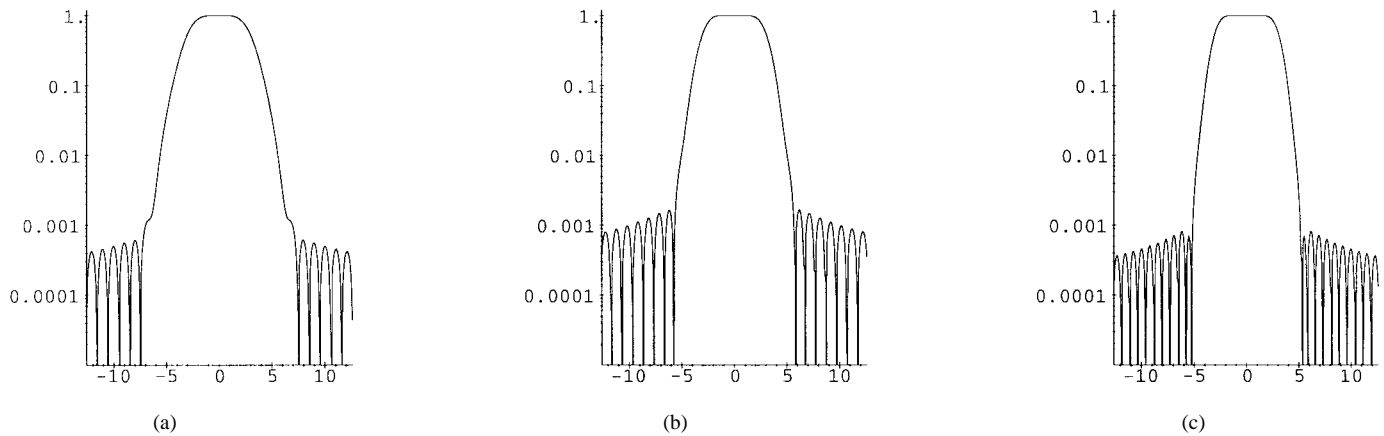


Fig. 26. Logarithmic plots of the Fourier magnitude for truncated Gaussian kernels. (a) Gaussian second-order, $N = 6$. (b) Gaussian sixth-order, $N = 6$. (c) Gaussian tenth-order, $N = 8$.

IV. RESULTS

These kernels have been compared on various images including situations typically encountered in medical applications. In each case, the efficiency and accuracy of a particular interpolation technique was evaluated by analyzing its Fourier properties, visual quality, interpolation error, complexity, and runtime.

A. Fourier Analysis

The Fourier properties of each method have been described in the previous sections. Figs. 5–24 allow the comparison of the interpolation kernels in both spatial and Fourier domains. For quality assessment in the Fourier domain, we have focused on three characteristics: 1) deviation from the ideal constant gain within the passband; 2) the amplitude and slope of the kernel's Fourier transform at the cutoff frequency; and 3) the occurrence and the amplitudes of ripples and sidelobes in the stopband. Deviation within the passband is important because attenuation within the passband causes blurring, while amplification improves the interpolated image's sharpness along with image noise. The importance of amplitude at cutoff frequency stems from the fact that high cutoff amplitudes in small slopes cause aliasing effects. Sidelobe anomalies can be significant because they alias the repetitions of the discrete image spectrum into the passband. Note that the importance of each criterion depends on the Fourier spectrum of the image to be interpolated. The ideal mask yields a rectangular Fourier shape with constant amplification in the passband, infinite slope at cutoff frequency, and zero values in the stopband (Fig. 5). However, truncations in the spatial domain result in notable overshoots in the passband and extensive ripples in the stopband (Figs. 6 and 7).

1) *Passband*: Nearest neighbor and linear interpolation, as well as quadratic and cubic B-spline approximation and the notch filter, show the largest deviation from the ideal rectangular shape in the passband (Figs. 9, 10, 11, 13, and 19, respectively). Therefore, images will be strongly smoothed during interpolation, and these methods only should be used for scenes without sharp edges and high local contrasts. The best passband characteristics are provided by windowed

sinc and cubic B-spline interpolations, as well as cubic, and Lagrange interpolations with large kernel sizes and all Gaussian kernels (Figs. 8, 14, 17, 21, 22, and 24, respectively).

2) *Cutoff Point*: The notch filter produces the best cutoff performance (Fig. 19). Truncated sinc and cubic B-spline interpolations, as well as cubic, Lagrange, and Gaussian kernels with $N > 4$ are suitable also (Figs. 6, 7, 14, 17, 21, and 24, respectively). The worst cutoff performance is shown by the nearest neighbor interpolator (Fig. 9). The use of linear interpolation and the Mitchell and Netravali's subjective best method should be avoided also regarding the cutoff criterion (Figs. 10 and 18, respectively).

3) *Stopband*: Similar to the kernels' gain at the cutoff point, the stopband characteristics are responsible for aliasing and Moiré effects. Truncated sinc, nearest neighbor, linear, and quadratic interpolation, the 4×4 cubic interpolation ($a = -1.3$), the notch filter, and the Lagrange interpolators produce ripples or sidelobes with amplitudes larger than 1% (Figs. 6, 7, 9, 10, 12, 15, 19–21, respectively). The ripples of the truncated Gaussian kernels are below 0.1% (Fig. 26) and those of the Blackman–Harris windowed sinc interpolator are even below 0.01% (Fig. 8).

Summarizing the Fourier analysis of passband, stopband, and cutoff frequency, the nearest neighbor and linear interpolations should be avoided, while the preferred method is the Gaussian kernel with large sizes.

B. Interpolation Quality

The sharply focused photograph of a human eye (Fig. 1) is interpolated when correcting the aspect ratio. After initial histogram stretching, the aspect ratio correction was performed by each of the interpolation methods. To visualize the interpolation error, the aspect-ratio-corrected image was interpolated again for downsizing to its initial size. The same interpolation method was used for both forward and backward transformation. This approach was favored over that suggested by Unser *et al.*, which advocates the use of loops rather than reciprocal transforms because the loop approach works only for certain applications, e.g., 16 successive rotations of 22.5° [14].

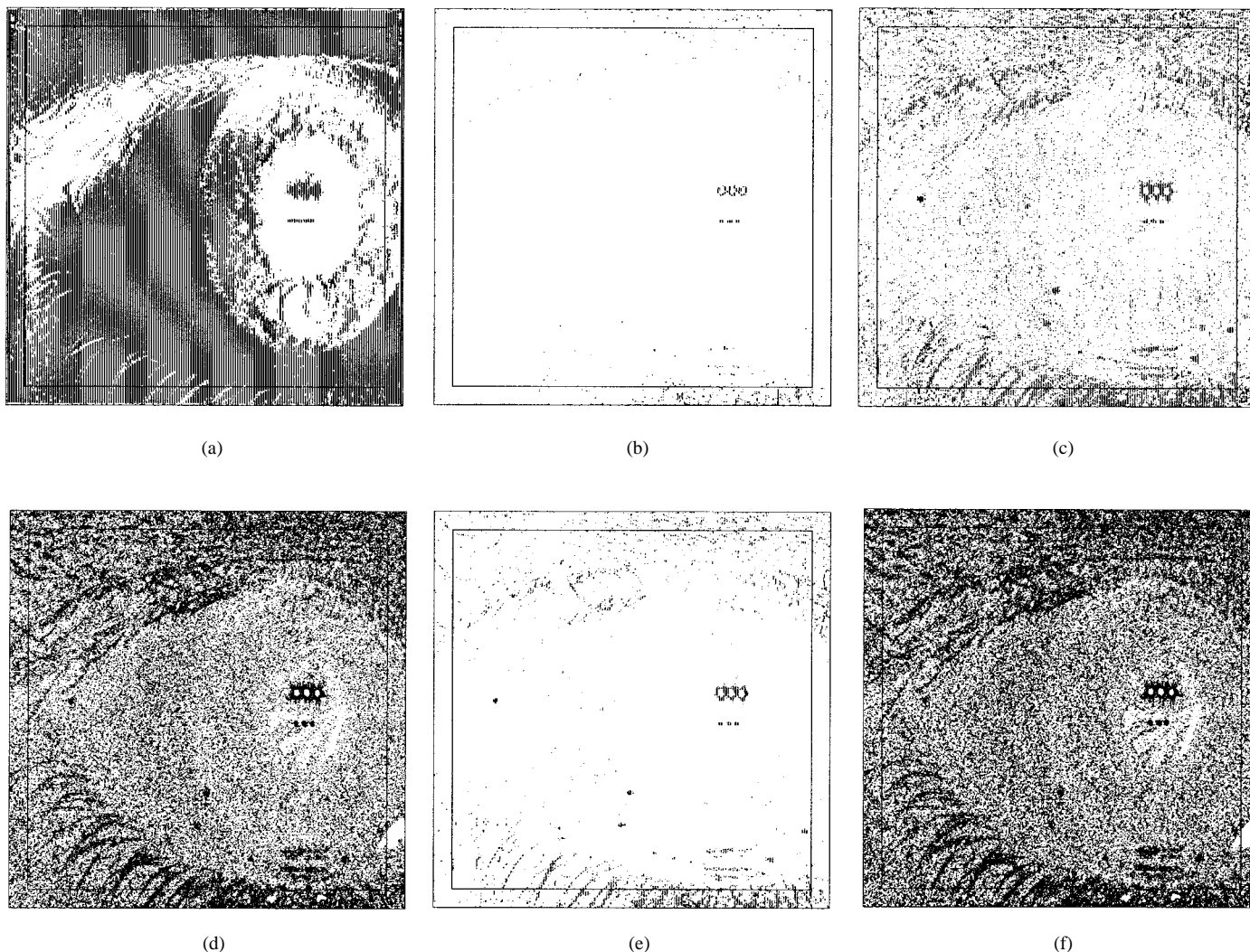


Fig. 27. Results of aspect ratio correction of the photograph. Pixels with error > 1.0 are displayed in black. The frame indicates the inner area for quantitative error comparisons ignoring border effects. (a) truncated sinc, $N = 7$. (b) Blackman-Harris windowed sinc, $N = 6$. (c) Linear interpolation. (d) Quadratic approximation. (e) Quadratic interpolation. (f) Cubic B-spline approximation.

The interpolation quality is assessed by the pixelwise absolute difference before and after the successive interpolations. The subtractions in Fig. 27 appreciably demonstrate the qualitative difference in interpolation quality performed with each method. All pixels that differ by more than one grey scale unit after forward and successive backward transformation are shown in black, while all others are displayed in white.

1) Interpolation Versus Approximation: In Fig. 27, interpolators versus approximators can be recognized. Because every third column is reproduced exactly by real interpolation, the error images must show every third vertical image line in white. Contrarily, approximators modify all pixels. Fig. 27(d), (f), (o), and (p) obviously identifies approximation methods: the quadratic and cubic B-spline approximators, the Mitchell and Netravali's subjective best, and the notch filter, respectively. In the previous section, the Gaussian kernels are shown not to be real interpolators. Especially those kernels with small orders M fail to fit exactly the zero points of the ideal sinc function. This effect is verified by inspecting

Fig. 27(t), (u), and (v) from Gaussian $(M, N) = (2, 4)$, $(2, 6)$, and $(6, 6)$, respectively.

2) DC-Constancy Versus DC-Inconstancy: By inspecting the subtraction images in Fig. 27, DC-constant versus DC-inconstant kernels are also differentiable. The eight-bit values of the eye image (Fig. 1) range from 0 (black) to 255 (white). Since DC-inconstancy usually affects high grey values more than lower ones, the interpolation error by DC-inconstant kernels is concentrated in bright image regions. From that point of view, Fig. 27(a) and (t)–(x) corresponds to DC-inconstant kernels, the truncated sinc as well as the Gaussian family, respectively.

For most common applications, approximators as well as DC-inconstant kernels produce poor results. Quality differences in DC-constant interpolators are indicated by the number of black pixels. Fig. 27(c), (h), and (l) shows many error pixels in linear, two-point cubic, and four-point cubic interpolations with $a = -1.3$, respectively. Note the latter kernel's overshoot in the passband [Fig. 15(b)]. Nearest neighbor interpolation

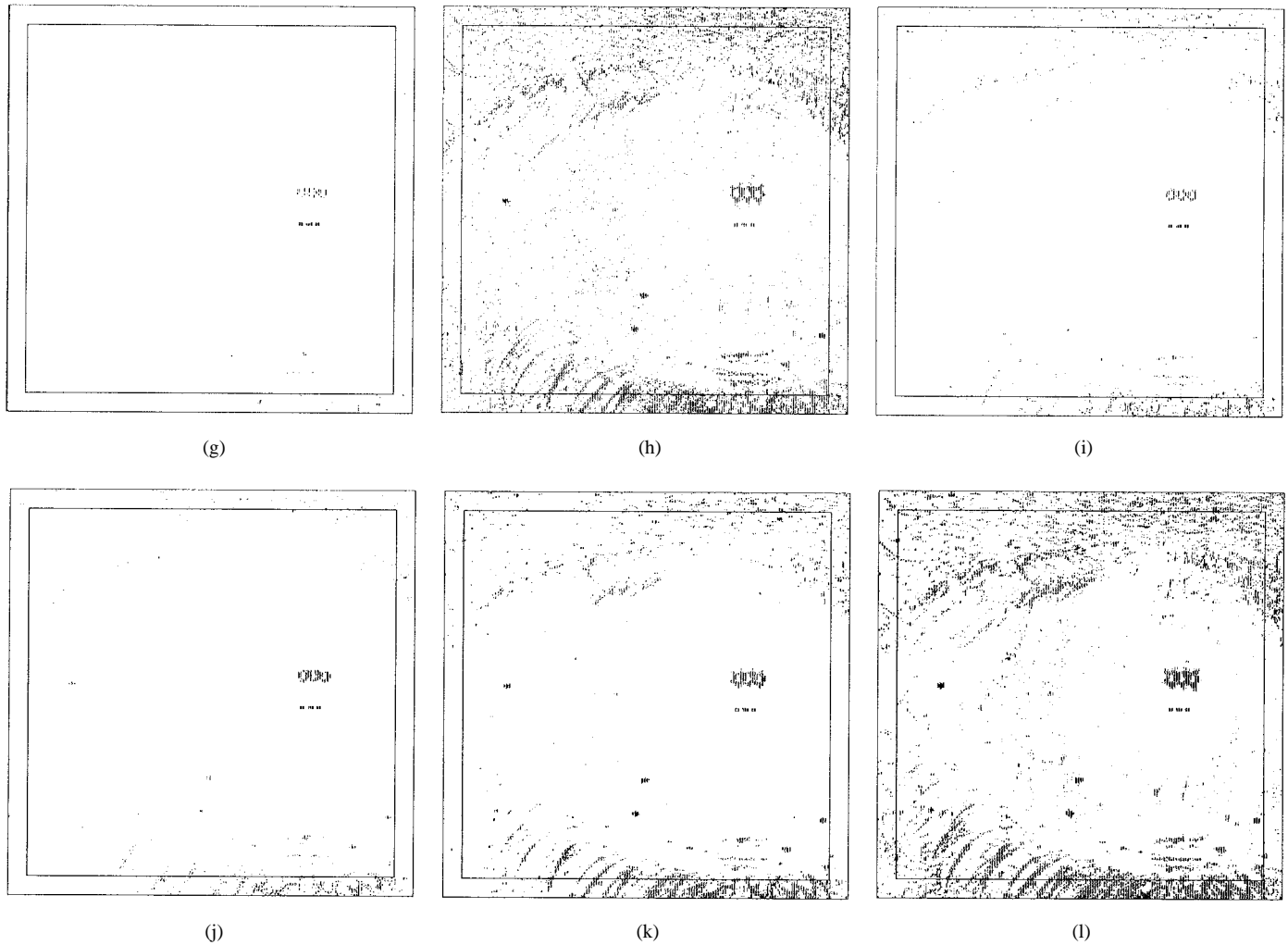


Fig. 27. (Continued.) Results of aspect ratio correction of the photograph. Pixels with error >1.0 are displayed in black. The frame indicates the inner area for quantitative error comparisons ignoring border effects. (g) Cubic B-spline interpolation. (h) Cubic interpolation, $N = 2$. (i) Cubic interpolation, $N = 4$, $a = -1/2$. (j) Cubic interpolation, $N = 4$, $a = -3/4$. (k) Cubic interpolation, $N = 4$, $a = -1$. (l) Cubic interpolation, $N = 4$, $a = -1.3$.

would even be worse. However, in this particular task, the nearest neighbor method exactly reproduces the image because the interpolation error introduced with the first transform is compensated exactly by the second. Therefore, nearest neighbor is not displayed in Fig. 27.

The $N =$ four-point cubic interpolators often are used in image processing. Fig. 27(i)–(l) compares the interpolation error for the parameter a descending from $-1/2$, $-3/4$, -1 , to -1.3 , respectively. For $a = -3/4$, the lowest number of error pixels is shown. Note that this parameter is obtained if the cubic kernel is forced to be C2-continuous. The same effect is observed for $N =$ six point cubics. In this experiment, the C2-continuous kernel [Fig. 27(n)] is superior to that proposed by Keys [Fig. 27(m)]. The overall smallest error values are indicated by nearly white error images with only a few labeled pixels: the cubic B-spline interpolation, as well as the C2-continuous cubics and the Lagrange interpolators, each with large kernel sizes [Figs. 27(g), (n), and (r)–(s), respectively].

C. Quantitative Error Analysis

The interpolation quality was quantitatively compared using a mathematical similarity measure. Again, the original image

and the one obtained after computing a forward and backward geometric transform, $s(k, l)$ and $r(k, l)$, respectively, were used for evaluation. According to our previous work, the normalized cross-correlation coefficient C was used to assess image similarity [21]

$$C = \left| \frac{\sum_{k,l} s(k, l)r(k, l) - KL\bar{s}\bar{r}}{\sqrt{(\sum_{k,l} s^2(k, l) - KL\bar{s}^2)(\sum_{k,l} r^2(k, l) - KL\bar{r}^2)}} \right| \quad (34)$$

where \bar{s} and \bar{r} denote the mean of the original and the twice interpolated image of the dimensions $K \times L$, respectively.

To avoid border effects, the centered subimages within a 25-pixel frame have been extracted before (34) was computed. In Fig. 27, this border is indicated by the rectangle. The border width was chosen with respect to the B-spline kernel. In contrast to all other methods discussed in this paper, the resulting B-spline interpolation kernel $\text{Spline}_h(x)$ is infinite. However, $\text{Spline}_h(x) < 10^{-7}$ for $x > x_d = 12$. Because the number of pixels is halved by the perspective transform of our third experiment, the frame width was selected to be $2 \cdot x_d$. This size has also been confirmed experimentally.

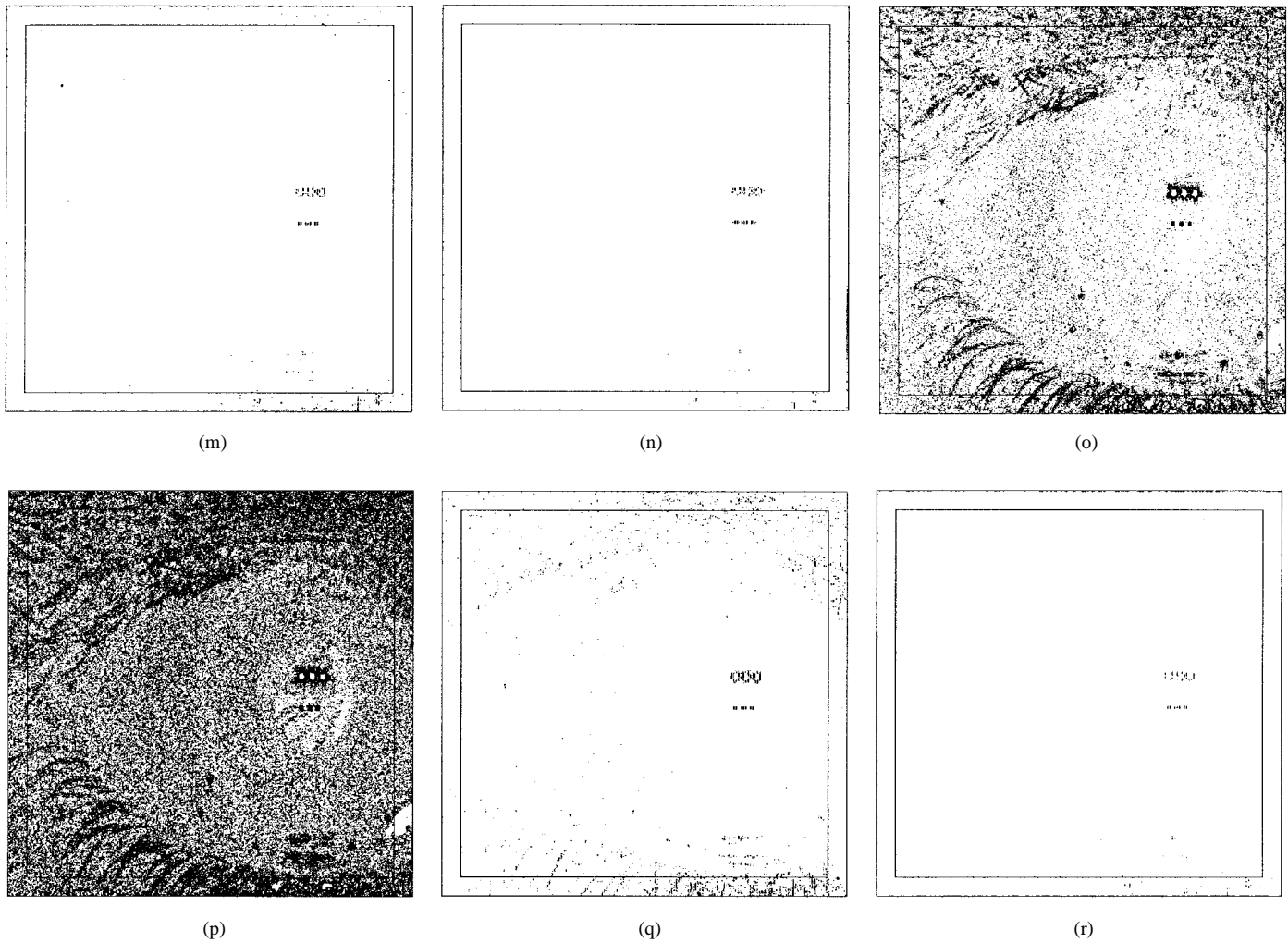


Fig. 27. (Continued.) Results of aspect ratio correction of the photograph. Pixels with error > 1.0 are displayed in black. The frame indicates the inner area for quantitative error comparisons ignoring border effects. (m) Cubic interpolation, $N = 6$ (Keys). (n) Cubic interpolation, $N = 6$ (C2-cont.). (o) Mitchell and Netravali's subjective best, $N = 4$. (p) Notch filter, $N = 4$. (q) Lagrange interpolation $N = 4$. (r) Lagrange interpolation $N = 5$.

In most medical applications of interpolation, the geometric transform is rather moderate and the number of pixels containing structured information may be reduced only slightly after interpolation. Therefore, the correlation coefficients C are expected to be nearly 1.0 for all methods. To rank the various interpolation methods a linear score is computed. According to the kernel size, linear interpolation is scored zero and cubic B-spline interpolation is scored one

$$S_C = \frac{\text{Mean}(C) - \text{Mean}(C_{\text{linear}})}{\text{Mean}(C_{\text{spline}}) - \text{Mean}(C_{\text{linear}})}. \quad (35)$$

Three common tasks of image interpolation in medical imaging have been introduced in Section II for the experimental comparison of interpolation methods. The first experiment was designed to quantify the qualitative results obtained in Section IV.B (Interpolation quality). However, experimental results depend on both the content of the scene and the geometric transform, which are analyzed by the second and third experiments, respectively.

1) *Correction of Aspect Ratios in CCD-Photographs:* Table II shows the similarities C and scores S_C obtained by the aspect ratio correction of the eye image (Fig. 1).

As mentioned before, the nearest neighbor method exactly reproduces the image in this specific task. The kernel's CO-discontinuities at $|x| = 1/2$ result in a shift of the pixel values. In the case of aspect ratio expansion, this shift is backtracked during the restoration of the initial image size. Therefore, no score is given for the nearest neighbor kernel. The odd Lagrange kernels also show CO-discontinuities at $|x| = 1/2$. Like the nearest neighbor effect incorporated with this specific transform, the scores $S_C > 0.99$ of the odd Lagrange kernels do not reflect the general method's quality.

The cubic B-spline interpolator's quality also is obtained by the C2-continuous cubic interpolators with $N = 8$ and $N = 6$ points. Note that the continuity of the second derivation of polynomials also results in the best 4×4 kernel. However, nearly half of the improvement from the $N = 2$ linear FIR-kernel to the $N \rightarrow \infty$ B-spline IIR-interpolator already is obtained by the 2×2 cubic interpolation. All approximators and the DC-inconstant truncated sinc interpolators result in worse than linear interpolation, independent of the size of the kernel. In contrast to the excellent Fourier properties, the Gaussian kernels are not convincing in this experiment, which

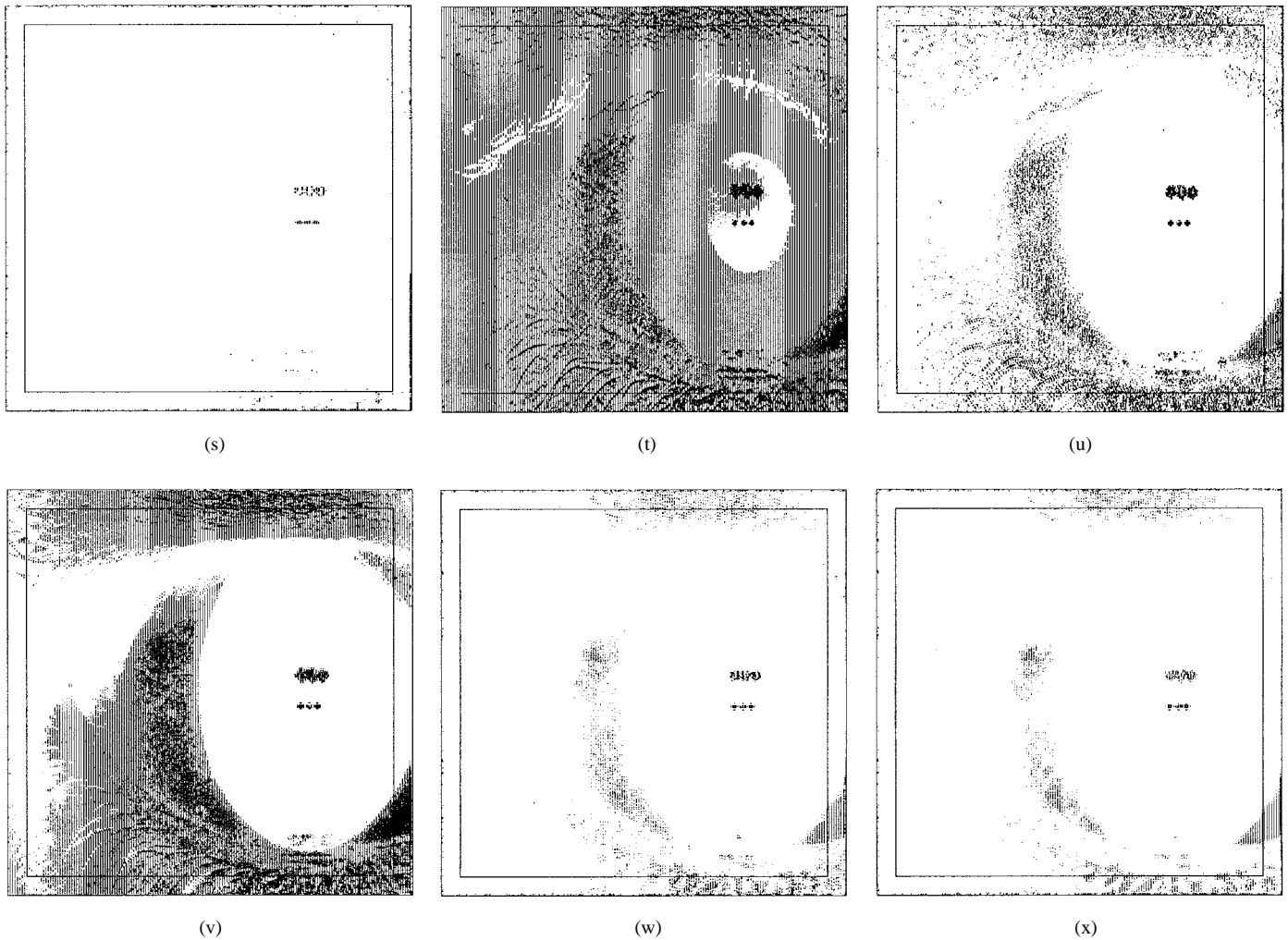


Fig. 27. (Continued.) Results of aspect ratio correction of the photograph. Pixels with error > 1.0 are displayed in black. The frame indicates the inner area for quantitative error comparisons ignoring border effects. (s) Lagrange interpolation, $N = 8$. (t) Second-order Gaussian interpolation, $N = 4$. (u) Second-order Gaussian interpolation, $N = 6$. (v) Sixth-order Gaussian interpolation, $N = 6$. (w) Sixth-order Gaussian interpolation, $N = 8$. (x) tenth-order Gaussian interpolation, $N = 8$.

is caused by both their insufficient interpolator characteristics and their DC-inconstancy.

2) *Rotation of MRI Sections:* The second interpolation task was performed by rotating the MR image (Fig. 2). In contrast to the aspect ratio correction, the number of pixels contributing to the image nearly is unchanged and almost all pixels must be recalculated for both forward and backward rotation. Therefore, the absolute value of the correlation coefficient is reduced in general. Nevertheless, the amount of image distortion caused by interpolation depends on the chosen angulation. Fixed angles of 22.5° , 30° , and 43° have been used for the comparison of interpolation methods in [14], [1], and [9], respectively. In this paper, 50 normal distributed angles ($N(\mu, \sigma) = N(45^\circ, 10^\circ)$) have been determined randomly and used for quantitative interpolation error evaluation. The mean correlation $\text{Mean}(C)$ of 50 corresponding forward and backward rotations is summarized in Table III and scored in the same fashion as in the previous example. The standard deviations $\text{StD}(C)$ verify the dependency of the current geometric transform and were used for p -value determination by the Student's t -test.

This time, only the tenth-order Gaussian kernel with $N = 8$ support points is superior significantly to the cubic B-spline interpolator ($p \ll 0.005$). The C2-continuous 8×8 cubic and the $N = 7$ -point Lagrange kernel have scores equal to the cubic B-spline interpolator and no significance was demonstrated. In general, differences in scores of at least 0.01 are required in Table III for statistical significance ($p < 0.005$). Hence, the C2-continuous cubic interpolator is best for $N = 6$ and $N = 4$ points. Note that slight variations of the free parameter a for the four-point cubic interpolation scheme raise the scores from negative values (even worse than linear interpolation) up to 90% of the cubic B-spline's IIR quality. The C2-continuous four-point cubic ($a = -3/4$) performs significantly better ($p \ll 0.005$) than all other cubic four-point kernels including those suggested by Keys.

3) *Perspective Projection of X-Ray Images:* The third experiment was designed to assess the influence of the test image selection on the interpolation errors. A set of 50 dental radiographs was arbitrarily chosen from clinical records. Some of these *in vivo* radiographs show teeth with or without fillings, while others show dental implants (Fig. 3). The radiographs

TABLE II
RESULTS OF ASPECT RATIO CORRECTION OF THE PHOTOGRAPH

Interpolation scheme	N	Order	C	S_C
Nearest neighbor	1×1	—	1.0000000	—
Linear	2×2	—	0.9999217	0
Cubic	2×2	$a = 0$	0.9999563	0.45
Quadratic (approx.)	3×3	$a = \frac{1}{2}$	0.9995775	< 0
Quadratic (interp.)	3×3	$a = 1$	0.9999771	0.72
Lagrange	3×3	—	0.9999972	0.99
B-spline (approx.)	4×4	cubic	0.9993620	< 0
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	4×4	$a = -\frac{1}{3}$	0.9999896	0.89
Cubic (C2-cont.)	4×4	$a = -\frac{1}{4}$	0.9999941	0.95
Cubic	4×4	$a = -1$	0.9999865	0.85
Cubic	4×4	$a = -1.3$	0.9999571	0.46
Cubic (approx.)	4×4	$b = \frac{1}{3}, c = \frac{1}{3}$	0.9998717	< 0
Cubic (notch filter)	4×4	$b = \frac{1}{2}, c = -\frac{1}{4}$	0.9989262	< 0
Lagrange	4×4	—	0.9999820	0.79
Gaussian	4×4	$M = 2$	0.9991061	< 0
Truncated sinc	5×5	—	0.9995952	< 0
Lagrange	5×5	—	0.9999985	1.01
Truncated sinc	6×6	—	0.9883882	< 0
Windowed sinc	6×6	—	0.9999926	0.93
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	6×6	—	0.9999949	0.96
Cubic (C2-cont.)	6×6	—	0.9999977	1.00
Lagrange	6×6	—	0.9999919	0.92
Gaussian	6×6	$M = 2$	0.9999580	0.47
Gaussian	6×6	$M = 6$	0.9999127	< 0
Truncated sinc	7×7	—	0.9998969	< 0
Lagrange	7×7	—	0.9999991	1.02
Cubic (C2-cont.)	8×8	—	0.9999979	1.00
Lagrange	8×8	—	0.9999955	0.97
Gaussian	8×8	$M = 6$	0.9999949	0.96
Gaussian	8×8	$M = 10$	0.9999815	0.78
B-spline (interp.)	—	cubic	0.9999979	1

were acquired from the jaws of both humans and dogs. Each image was projected perspectively both forward and backward. Again, the standard deviation of the interpolation error as well as its mean were determined and analyzed for all methods included in this study.

The results of the third experiment are presented in Table IV. Although the selected perspective transform approximately halves the number of pixels, the mean correlation is still about 1.0, but smaller in comparison to the former experiments. Since the standard deviation is enlarged, the results in general depend more on the image content than on the interpolation task. However, all kernels with $S_C > 0$ significantly outperform the linear interpolator except the two-point cubic ($S_C = 0.1$). The p -values are below 0.005 except for the 0.40-scored Mitchell and Netravali's subjective best kernel ($p < 0.05$). The Lagrange kernels with $N = 8$ and $N = 6$ points, the sixth-order Gaussian method with $N = 6$, and Keys' 6×6 cubic interpolator are scored larger than the cubic B-spline interpolator. Again, the C2-continuous cubics with $N = 8$ and $N = 6$ equal the performance of the cubic B-spline interpolator. Note that the differences among the large group of methods scored above 0.85 are not significant statistically. As with the other experiments, the C2-continuous cubic interpolator turned out to be the best cubic 4×4 mask. Significance was found to the $a = -1$ kernel ($p < 0.05$) and all other four-point cubics ($p \ll 0.005$), except the $a = -1/2$ kernel introduced by Keys. The C2-continuous

cubic kernel already yields 96% of the improvement obtained by the migration from linear to B-spline interpolation. As mentioned previously, it depends on the contents of the image whether there is a gain from using a four-point C2-continuous cubic kernel or the cubic B-spline IIR-interpolator.

D. Computational Complexity

The convolution of $N \times N$ mask points with the weights $h(x)$ of a separated kernel takes $N^2 + N$ multiplications, $N^2 - 1$ additions and $N^2 + 1$ memory accesses. The only differences in complexity between the various interpolation and approximation methods are expressed by the complexity of the kernel $h(x)$ and by the prefiltering step in the case of B-spline interpolation. Table V shows the complexities of the mere kernels $h(x)$ and the computation of the prefiltered image values necessary for B-spline interpolation for one dimension only. Some kernels additionally require the evaluation of mathematical functions such as sine or cosine, which can be efficiently implemented by library calls.

The nearest neighbor kernel itself needs no computations at all. Linear interpolation simply uses the distances as weights whose computations involve one addition in each dimension. The number of additions of the piecewise quadratic and cubic polynomials was reduced by specializing the kernel for the various distances. Instead of computing the distance to the current mask point first and then evaluating the polynomial, the polynomials' coefficients are adjusted such that all polynomials take the distance to the center of the mask as argument and no other distances need to be computed. This also reduces the number of terms in some polynomials, e.g., in the case of the one-parameter cubics. All polynomials are implemented using the Horner scheme. The 1-D quadratic kernel with a mask size of $N = 3$ requires six multiplications and five additions. The piecewise cubic kernels need $3N$ multiplications. Depending on the parameter choice, some coefficients are zero and the number of additions varies. If B-spline interpolation is intended, the image must be prefiltered. Unser *et al.* have developed a fast recursive prefiltering algorithm, which in one dimension only needs two multiplications and two additions [10]. Additionally, the coefficients t for convolution have to be scaled once. Furthermore for each pixel and each dimension, two values must be retrieved from memory and one intermediate result, as well as the new pixel value, must be stored. Compared with traditional matrix approaches such as LU factorization, this involves either fewer multiplications and additions (if the LU factorization is computed together with the adjusted image points) or fewer memory accesses (if the LU factorization is precomputed). Each Lagrange term is the product of $N - 1$ scaled distances from the interpolated point to the mask points with the distance to the current mask point missing. If one splits each term into two products of the distances left and right from the current position and computes all terms together, both subproducts can be determined in an iterative manner. In our implementation of the Gaussian kernel, all terms with the same power are combined, their coefficient is precomputed, and the Horner scheme is applied. This significantly reduced the number of operations (Table V).

TABLE III
RESULTS OF ROTATING THE MR IMAGE

Interpolation scheme	N	Order	Mean(C)	StD(C)	S_C
Nearest neighbor	1×1	—	0.9928903	$11.8470 \cdot 10^{-4}$	< 0
Linear	2×2	—	0.9979303	$0.23633 \cdot 10^{-4}$	0
Cubic	2×2	$a = 0$	0.9982730	$0.41216 \cdot 10^{-4}$	0.18
Quadratic (approx.)	3×3	$a = \frac{1}{2}$	0.9962795	$0.01756 \cdot 10^{-4}$	< 0
Quadratic (interpol.)	3×3	$a = 1$	0.9991039	$0.26889 \cdot 10^{-4}$	0.60
Lagrange	3×3	—	0.9996543	$0.28137 \cdot 10^{-4}$	0.88
B-spline (approx.)	4×4	cubic	0.9944113	$0.00773 \cdot 10^{-4}$	< 0
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	4×4	$a = -\frac{1}{2}$	0.9996622	$0.11956 \cdot 10^{-4}$	0.89
Cubic (C2-cont.)	4×4	$a = -\frac{3}{4}$	0.9996832	$0.09277 \cdot 10^{-4}$	0.90
Cubic	4×4	$a = -1$	0.9990500	$0.87344 \cdot 10^{-4}$	0.57
Cubic	4×4	$a = -1.3$	0.9971569	$3.12454 \cdot 10^{-4}$	< 0
Cubic (approx.)	4×4	$b = \frac{1}{3}, c = \frac{1}{3}$	0.9986348	$0.06956 \cdot 10^{-4}$	0.36
Cubic (notch filter)	4×4	$b = \frac{1}{2}, c = -\frac{1}{4}$	0.9902155	$0.03835 \cdot 10^{-4}$	< 0
Lagrange	4×4	—	0.9995642	$0.13466 \cdot 10^{-4}$	0.84
Gaussian	4×4	$M = 2$	0.9993863	$0.58101 \cdot 10^{-4}$	0.75
Truncated sinc	5×5	—	0.9966259	$5.25823 \cdot 10^{-4}$	< 0
Lagrange	5×5	—	0.9998303	$0.05710 \cdot 10^{-4}$	0.98
Truncated sinc	6×6	—	0.9927909	$8.93101 \cdot 10^{-4}$	< 0
Windowed sinc	6×6	—	0.9997279	$0.12405 \cdot 10^{-4}$	0.92
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	6×6	—	0.9998022	$0.11676 \cdot 10^{-4}$	0.96
Cubic (C2-cont.)	6×6	—	0.9998630	$0.07970 \cdot 10^{-4}$	0.99
Lagrange	6×6	—	0.9997688	$0.12720 \cdot 10^{-4}$	0.94
Gaussian	6×6	$M = 2$	0.9997001	$0.17202 \cdot 10^{-4}$	0.91
Gaussian	6×6	$M = 6$	0.9998250	$0.13613 \cdot 10^{-4}$	0.97
Truncated sinc	7×7	—	0.9978243	$3.50711 \cdot 10^{-4}$	< 0
Lagrange	7×7	—	0.9998782	$0.02683 \cdot 10^{-4}$	1.00
Cubic (C2-cont.)	8×8	—	0.9998767	$0.09077 \cdot 10^{-4}$	1.00
Lagrange	8×8	—	0.9998388	$0.11275 \cdot 10^{-4}$	0.98
Gaussian	8×8	$M = 6$	0.9998630	$0.10039 \cdot 10^{-4}$	0.99
Gaussian	8×8	$M = 10$	0.9998956	$0.08430 \cdot 10^{-4}$	1.01
B-spline (interpol.)	—	cubic	0.9998789	$0.09218 \cdot 10^{-4}$	1

E. Runtime Measurements

The runtimes of the various interpolation schemes were measured on a Sun Ultra 1. Sources have been compiled using GNU's gcc version 2.7 without optimization. A shell script was used to average 50 rotations of the MR image of the head. As the rotation is quite time consuming and hinders the comparison of the mere interpolation times, Fig. 28 distinguishes between interpolation and geometric transform time. The listed runtimes, however, refer to the entire task: interpolation and rotation.

The rotation of the 256×256 pixel image took approximately 0.10 s. Simple interpolation methods such as nearest neighbor, linear, and 2×2 cubic interpolation are fairly fast and require less time than the rotation of the pixel coordinates. Also the 3×3 kernels run quickly, both for quadratic and Lagrange interpolations, with 0.12 and 0.13 s, respectively. Compared with the 4×4 cubic kernel quadratic interpolation took 70.6% of the cubic interpolation time. Our theoretical examinations suggest a performance gain of approximately 60%. In general, the Lagrange kernels are a bit slower than the piecewise cubic polynomials of the same mask size. Interpolation with the cubic B-spline took about 1.5 times as long as interpolation with a 4×4 cubic kernel, but less time than interpolation with a 6×6 cubic kernel. The Gaussian interpolation required around 0.63, 1.10, and 1.54 s for mask sizes of 16, 36, and 64 pixels, respectively. This poor performance is caused by the evaluations of the exponential

function necessary to determine the weights. Increasing the order of the Gaussian kernel slows the interpolation only modestly. Even so, this is time consuming, and efficiency can be gained by precomputing the kernel's weights and storing them in a lookup table (LUT). Then, only the indexes are determined during interpolation and the weights to convolve with are retrieved from memory resulting in $2N$ memory accesses. The dotted lines in Fig. 28 indicate the limits given by 4×4 , 6×6 , and 8×8 LUT's filled with 10000 elements per unit. These sizes have been determined according to unchanged interpolation errors and correspond to those suggested by Ostuni *et al.* [26]. This is advantageous for computationally intensive kernels like the Gaussians or truncated sincs. However, direct computation of the cubic and Lagrange weights is faster than retrieving them from memory, as in the LUT approach.

If the image is only zoomed in or out, further accelerations can be achieved. Especially for rotations, Danielson and Hammerin, and Unser *et al.* describe fast separable algorithms which perform the rotation through successive 1-D distortions ([8], [9], and [14], respectively).

V. DISCUSSION

Although image interpolation is as old as computer vision, interpolation techniques are still discussed, and new techniques are introduced [11], [19]. Furthermore, different names refer to the same techniques, and several names are used redundantly

TABLE IV
RESULTS OF PERSPECTIVE PROJECTION OF X-RAY IMAGES

Interpolation scheme	N	Order	Mean(C)	StD(C)	S_C
Nearest neighbor	1×1	—	0.9936343	$19.0280 \cdot 10^{-4}$	< 0
Linear	2×2	—	0.9980701	$5.25685 \cdot 10^{-4}$	0
Cubic	2×2	$a = 0$	0.9981339	$5.04139 \cdot 10^{-4}$	0.10
Quadratic (approx.)	3×3	$a = \frac{1}{2}$	0.9974805	$6.98560 \cdot 10^{-4}$	< 0
Quadratic (interpol.)	3×3	$a = 1$	0.9984671	$4.19490 \cdot 10^{-4}$	0.62
Lagrange	3×3	—	0.9984819	$4.25232 \cdot 10^{-4}$	0.65
B-spline (approx.)	4×4	cubic	0.9968359	$9.02280 \cdot 10^{-4}$	< 0
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	4×4	$a = -\frac{1}{2}$	0.9986753	$3.78956 \cdot 10^{-4}$	0.95
Cubic (C2-cont.)	4×4	$a = -\frac{3}{4}$	0.9986815	$3.84236 \cdot 10^{-4}$	0.96
Cubic	4×4	$a = -1$	0.9984586	$4.28731 \cdot 10^{-4}$	0.61
Cubic	4×4	$a = -1.3$	0.9978173	$5.71267 \cdot 10^{-4}$	< 0
Cubic (approx.)	4×4	$b = \frac{1}{2}, c = \frac{1}{3}$	0.9983238	$4.56332 \cdot 10^{-4}$	0.40
Cubic (notch filter)	4×4	$b = \frac{1}{2}, c = -\frac{1}{4}$	0.9955231	$13.4080 \cdot 10^{-4}$	< 0
Lagrange	4×4	—	0.9986569	$3.82337 \cdot 10^{-4}$	0.92
Gaussian	4×4	$M = 2$	0.9974407	$9.61097 \cdot 10^{-4}$	< 0
Truncated sinc	5×5	—	0.9961569	$11.3030 \cdot 10^{-4}$	< 0
Lagrange	5×5	—	0.9986287	$4.01965 \cdot 10^{-4}$	0.88
Truncated sinc	6×6	—	0.9815888	$87.2910 \cdot 10^{-4}$	< 0
Windowed sinc	6×6	—	0.9986980	$3.76893 \cdot 10^{-4}$	0.99
Cubic, $h \propto \mathcal{T}\{\text{sinc}\}$	6×6	—	0.9987203	$3.76786 \cdot 10^{-4}$	1.02
Cubic (C2-cont.)	6×6	—	0.9987081	$3.87584 \cdot 10^{-4}$	1.00
Lagrange	6×6	—	0.9987273	$3.73975 \cdot 10^{-4}$	1.03
Gaussian	6×6	$M = 2$	0.9986155	$4.08325 \cdot 10^{-4}$	0.86
Gaussian	6×6	$M = 6$	0.9986203	$4.18948 \cdot 10^{-4}$	0.87
Truncated sinc	7×7	—	0.9972603	$7.51157 \cdot 10^{-4}$	< 0
Lagrange	7×7	—	0.9986487	$4.03737 \cdot 10^{-4}$	0.91
Cubic (C2-cont.)	8×8	—	0.9987059	$3.89373 \cdot 10^{-4}$	1.00
Lagrange	8×8	—	0.9987291	$3.79463 \cdot 10^{-4}$	1.04
Gaussian	8×8	$M = 6$	0.9987178	$3.84378 \cdot 10^{-4}$	1.02
Gaussian	8×8	$M = 10$	0.9986632	$4.07097 \cdot 10^{-4}$	0.93
B-spline (interpol.)	—	cubic	0.9987056	$3.89628 \cdot 10^{-4}$	1

TABLE V
COMPUTATIONAL COMPLEXITY

Interpolation scheme	Size	Multiplications	Additions	Others
Truncated sinc	N	$N + 2$	$N - 1$	\sin
Windowed sinc	N	$6N + 2$	$3N - 1$	$\sin + 2N \cos$
Nearest neighbor	1	—	—	—
Linear	2	—	1	—
Quadratic	3	6	5	—
B-spline (approx.)	4	12	8	—
interpol. prefilter	K	2	2	4 mem. access
Cubic, 1 param.	$N = 2, 4, 6, 8$	$3N$	$2N - 1$	—
Cubic, 2 param.	4	12	9	—
Lagrange	N	$4N - 6$	$2N - 3$	—
Gaussian	N	$N(\frac{M}{2} + 5)$	$N(\frac{M}{2} + 2) - 1$	$N \exp$

The number of operations are given per pixel. K , N , and M denote the number of samples of the 1D-signal, the size, and the order of the kernel, respectively.

to describe different methods. Therefore, one might have trouble in finding the optimal kernel for a specific interpolation application.

This paper compares the most commonly used interpolation techniques, including FIR-kernels with sizes of 1×1 to 8×8 and the cubic B-spline IIR-interpolator. Here, we did not differentiate between the processes of decreasing and increasing the data rate, which sometimes is called decimation versus interpolation, respectively [4]. Instead, we define the general characteristics of approximator versus interpolator [11] and of DC-constancy versus DC-inconstancy. The interpolation error increases with the more severe geometric deformations and depends on the content of the image. If the

number of image pixels before and after the interpolation is comparable, the interpolation error is expected to be smaller. All interpolation methods smooth the image more or less. Images with sharp-edged details and high local contrast are more affected by interpolation than others. The comparison is accomplished by spatial and Fourier analysis of the kernel functions, visual quality assessment, quantitative interpolation error determination, computational complexity analysis, and run time measurement based on representative applications and clinical images.

For each interpolation technique discussed in this paper, examples can be given where each scheme is optimal. In the following, each method's key features are stressed. In the

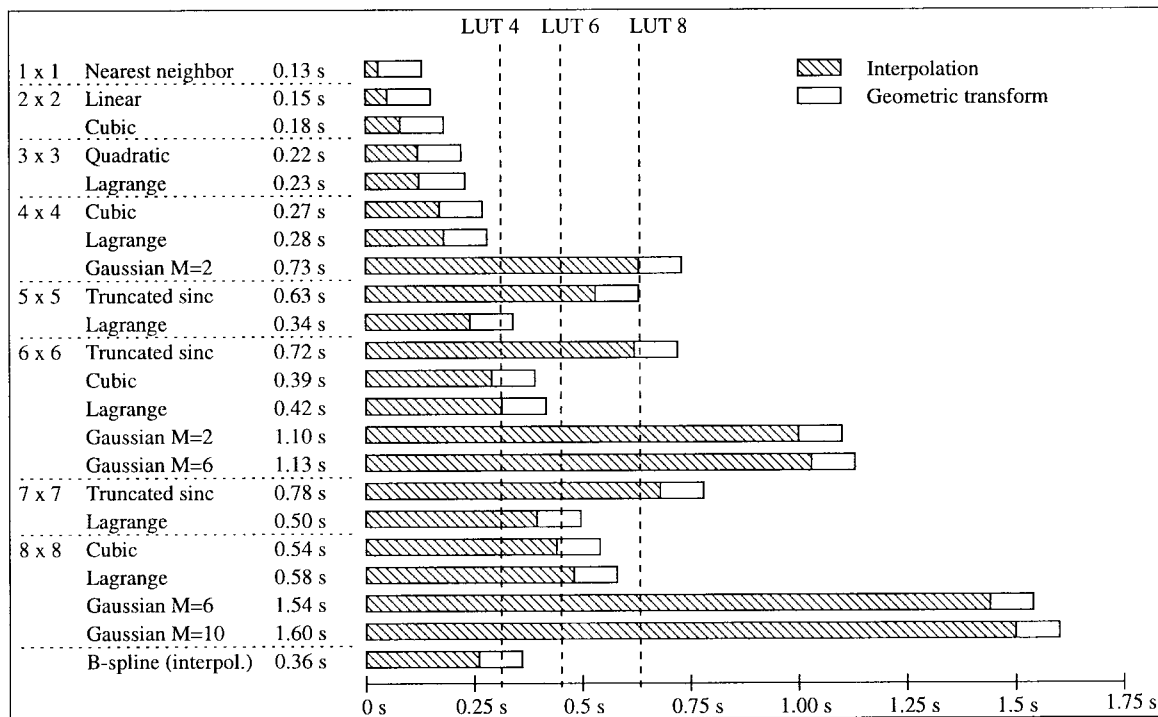


Fig. 28. Run times measured on a Sparc Ultra 1. Further speedup of the Gaussian and truncated sinc kernels is achieved by implementation via look up tables (LUT).

frequency space, the ideal interpolation method is represented by the IIR sinc kernel. Simple truncations of the infinite sinc function result in DC-inconstant interpolators, but proper windowing enables the design of high quality kernels. The Blackman–Harris windowed sinc yields the best stopband response in the Fourier domain. The nearest neighbor method is the fastest technique, but also incurs the largest interpolation error. The linear kernel requires no further multiplications. Although it has limited features, linear interpolation was recently discovered by Grevera and Udupa to be the most frequently mentioned method in publications during the past five years [34]. However, with just a little increase of computation time the 2×2 cubic kernel performs better when compared with the linear interpolator in all the quantitative experiments of this study. If the situation requires small kernel sizes, the two-point cubic method should be chosen, which confirms Maeland's conclusion [6].

In general, large kernel sizes were found to be superior to small interpolation masks. Although modern computers are able to process a huge amount of data in realtime, fast methods might be required for online resampling of image sequences or films [15]. Using quadratic 3×3 instead of cubic 2×2 interpolations, the interpolation error is decreased further. Our experiments show the three-point Lagrange technique to have smaller errors than the three-point quadratics, but odd Lagrange kernels are CO-discontinuous while even kernels are C1-discontinuous, which might be a knock-out criterion in some applications.

Several approaches to the design of four-point cubic kernels could be found in the literature. The two parameter approach [31], [33] comprises interpolators as well as approximators.

The notch filter is the best kernel with respect to the cutoff frequency characteristics, but it is weak in practice. Wolberg claims that despite the added flexibility made possible by a second parameter, the benefits of the two-parameter method compared to the one-parameter case should be scrutinized [13]. Ultimately, the authors suggest that merging image restoration with reconstruction can yield significant improvements in the quality of reconstruction filters.

From our numerical simulations, the C2-continuous cubics seem superior to other cubic approaches. In addition, the C2-continuity yields a generic concept suitable for the design of a family of kernels with different sizes. There are some interpolation tasks where the kernel's dimensions should be locally adaptive, e.g., transforming cartesian into polar coordinates on a rigid grid. Based on their definitions, Gaussian, Lagrange, and C2-continuous cubic interpolations seem to be suitable for such generic kernel families.

The B-spline interpolation method differs from all other techniques in this study. Since the B-spline approximator is not applied directly to the image data, its corresponding kernel is unlimited spatially. In other words, the B-spline interpolator is an IIR-filter. It produces one of the best results in terms of similarity to the original image, and of the top methods, it runs the fastest. Theory, efficient design, and applications of B-spline signal processing are analyzed by Unser, Aldroubi, and Eden [35]–[37]. They found the third-order (cubic) B-spline interpolator to be sufficient for several practical applications [10]. Further enlargement of the kernel's order will not only improve the interpolation quality, but also increase the numerical complexity of the kernel and the prefilter. In addition, this will magnify the edge effects, which

already are considerable when compared to FIR methods.

The Gaussian method introduced by Appledorn was designed to have excellent Fourier properties [19]. Using large kernels, the interpolation error was found to be quite small. However, the sum of the samples of the Gaussian kernel is not necessarily one, and the zero crossings do not fit exactly. Since those properties are essential for interpolation, artifacts are introduced, e.g., during aspect ratio correction of the eye image (see Fig. 27). Therefore, at the expense of some minor irregularities in the Fourier domain, the Gaussian method might be substantially improved if the kernels were forced to be DC-constant interpolants and agree with (5) and (6).

ACKNOWLEDGMENT

The authors would like to thank H.-G. Gröndahl, Göteborg University, Göteborg, Sweden, for providing the radiographs for this research. Many thanks also to S. and R. Webber, Wake Forest University, Winston-Salem, NC, for final improvements.

REFERENCES

- [1] T. Lehmann, W. Oberschelp, E. Pelikan, and R. Repges, *Bildverarbeitung für die Medizin: Grundlagen, Modelle, Methoden, Anwendungen* (in German). Berlin, Germany: Springer-Verlag, 1997.
- [2] S. W. Rowland, "Computer implementation of image reconstruction formulas," in *Image Reconstruction from Projections: Implementation and Applications*, G. T. Herman Ed. Berlin, Germany: Springer-Verlag, 1979, pp. 9–70.
- [3] J. D. Faires and R. L. Burden, *Numerical Methods*. Boston, MA: PWS, 1993.
- [4] H. S. Hou and H. C. Andrews, "Cubic splines for image interpolation and digital filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, no. 6, pp. 508–517, 1978.
- [5] R. G. Keys, "Cubic convolution interpolation for digital image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, no. 6, pp. 1153–1160, 1981.
- [6] E. Maelland, "On the comparison of interpolation methods," *IEEE Trans. Med. Imag.*, vol. MI-7, pp. 213–217, 1988.
- [7] J. A. Parker, R. V. Kenyon, and D. E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Trans. Med. Imag.*, vol. MI-2, pp. 31–39, 1983.
- [8] P. E. Danielsson and M. Hammerin, "High accuracy rotation of images," Department of Electrical Engineering, Linköping University, Sweden, Tech. Rep. LiTH-ISY-I-1152/1990.
- [9] P. E. Danielsson and M. Hammerin, "Note: High accuracy rotation of images," *CVGIP: Graph. Models Image Processing*, vol. 54, no. 4, pp. 340–344, 1992.
- [10] M. Unser, A. Aldroubi, and M. Eden, "Fast B-splines transforms for continuous image representation and interpolation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 13, pp. 277–285, 1991.
- [11] N. A. Dodgson, "Quadratic interpolation for image resampling," *IEEE Trans. Image Processing*, vol. 6, pp. 1322–1326, 1997.
- [12] A. Schaum, "Theory and design of local interpolators," *CVGIP: Graph. Models Image Processing*, vol. 55, no. 6, pp. 464–481, 1993.
- [13] G. Wolberg, *Digital Image Warping*. Los Alamitos, CA: IEEE Computer Society, 1990.
- [14] M. Unser, P. Thévenaz, and L. Yaroslavsky, "Convolution-based interpolation for fast, high-quality rotation of images," *IEEE Trans. Image Processing*, vol. 4, pp. 1371–1381, Oct. 1995.
- [15] O. D. Evans and Y. Kim, "Efficient implementation of image warping on a multimedia processor," *Real-Time Imag.*, vol. 4, pp. 417–428, 1998.
- [16] R. R. Schultze and R. L. Stevenson, "A Bayesian approach to image expansion for improved definition," *IEEE Trans. Image Processing*, vol. 3, no. 3, pp. 233–242, Mar. 1994.
- [17] S. Thurnhofer and S. Mitra, "Edge-enhanced image zooming," *Opt. Eng.*, vol. 35, no. 7, pp. 1862–1870, 1996.
- [18] G. J. Grevera, J. K. Udupa, and Y. Miki, "A task-specific evaluation of three-dimensional image interpolation techniques," *IEEE Trans. Med. Imag.*, vol. 18, pp. 137–143, Feb. 1999.
- [19] C. R. Appledorn, "A new approach to the interpolation of sampled data," *IEEE Trans. Med. Imag.*, vol. 15, pp. 369–376, 1996.
- [20] C. Lee, M. Eden, and M. Unser, "High-quality image resizing using oblique projection operators," *IEEE Trans. Image Processing*, vol. 7, no. 5, pp. 679–692, May, 1998.
- [21] T. Lehmann, A. Sovakar, W. Schmitt, and R. Repges, "A comparison of similarity measures for digital subtraction radiography," *Comp. Biol. Med.*, vol. 27, no. 2, pp. 251–257, 1997.
- [22] P. V. Sankar and L. A. Ferrari, "Simple algorithms and architecture for B-spline interpolation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-10, pp. 271–276, 1988.
- [23] T. Lehmann, A. Kaupp, R. Effert, and D. Meyer-Ebrecht, "Automatic strabometry by Hough-transformation and covariance-filtering," in *Proceedings ICIP-94*. Los Alamitos, CA: IEEE Computer Society, 1994, vol. 1, pp. 421–425.
- [24] J. West, M. Fitzpatrick, M. Y. Wang, B. M. Dawant et al., "Comparison and evaluation of retrospective intermodality brain image registration techniques," *J. Comp. Assisted Tomogr.*, vol. 21, no. 4, pp. 554–566, 1997.
- [25] T. Lehmann, H. G. Gröndahl, K. Gröndahl, W. Schmitt, and K. Spitzer, "Observer-independent registration technique of perspective projection prior to subtraction of in-vivo radiographs," *Dentomaxillofacial Radiol.*, vol. 27, no. 3, pp. 140–150, 1998.
- [26] J. L. Ostuni, A. K. S. Santha, V. S. Mattay, D. R. Weinberger, R. L. Levin, and J. A. Frank, "Analysis of interpolation effects in the reslicing of functional MR images," *J. Comp. Assisted Tomogr.*, vol. 21, no. 5, pp. 803–810, 1997.
- [27] F. J. Harris, "On the use of windows for harmonic analysis with the discrete Fourier-transform," *Proc. IEEE*, vol. 66, pp. 51–83, 1978.
- [28] R. W. Schafer and L. R. Rabiner, "A digital signal processing approach to interpolation," *Proc. IEEE*, vol. 61, pp. 692–702, 1973.
- [29] I. Her and C. T. Yuan, "Resampling on a pseudo-hexagonal grid," *CVGIP: Graph. Models Image Processing*, vol. 56, no. 4, pp. 336–347, 1994.
- [30] S. K. Park and R. A. Schowengerdt, "Image reconstruction by parametric cubic convolution," *CVGIP*, vol. 23, pp. 258–272, 1983.
- [31] S. E. Reichenbach and S. K. Park, "Two-parameter cubic convolution for image reconstruction," *Proc. SPIE*, vol. 1199, pp. 833–840, 1989.
- [32] N. A. Dodgson, "Image resampling," Univ. Cambridge Comp. Lab., Cambridge, U.K., Tech. Rep. 261, 1992.
- [33] D. P. Mitchell and A. N. Netravali, "Reconstruction filters in computer graphics," *Comp. Graph.*, vol. 22, no. 4, pp. 221–228, 1988.
- [34] G. J. Grevera and J. K. Udupa, "An objective comparison of 3-D image interpolation methods," *IEEE Trans. Med. Imag.*, vol. 17, no. 4, pp. 642–652, 1998.
- [35] M. Unser, A. Aldroubi, and M. Eden, "Polynomial spline signal approximations: Filter design and asymptotic equivalence with Shannon's sampling theorem," *IEEE Trans. Inform. Theory*, vol. 38, pp. 95–103, 1992.
- [36] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part I—Theory," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 821–833, 1993.
- [37] M. Unser, A. Aldroubi, and M. Eden, "B-spline signal processing: Part II—Efficient design and applications," *IEEE Trans. Signal Processing*, vol. 41, no. 2, pp. 834–848, 1993.