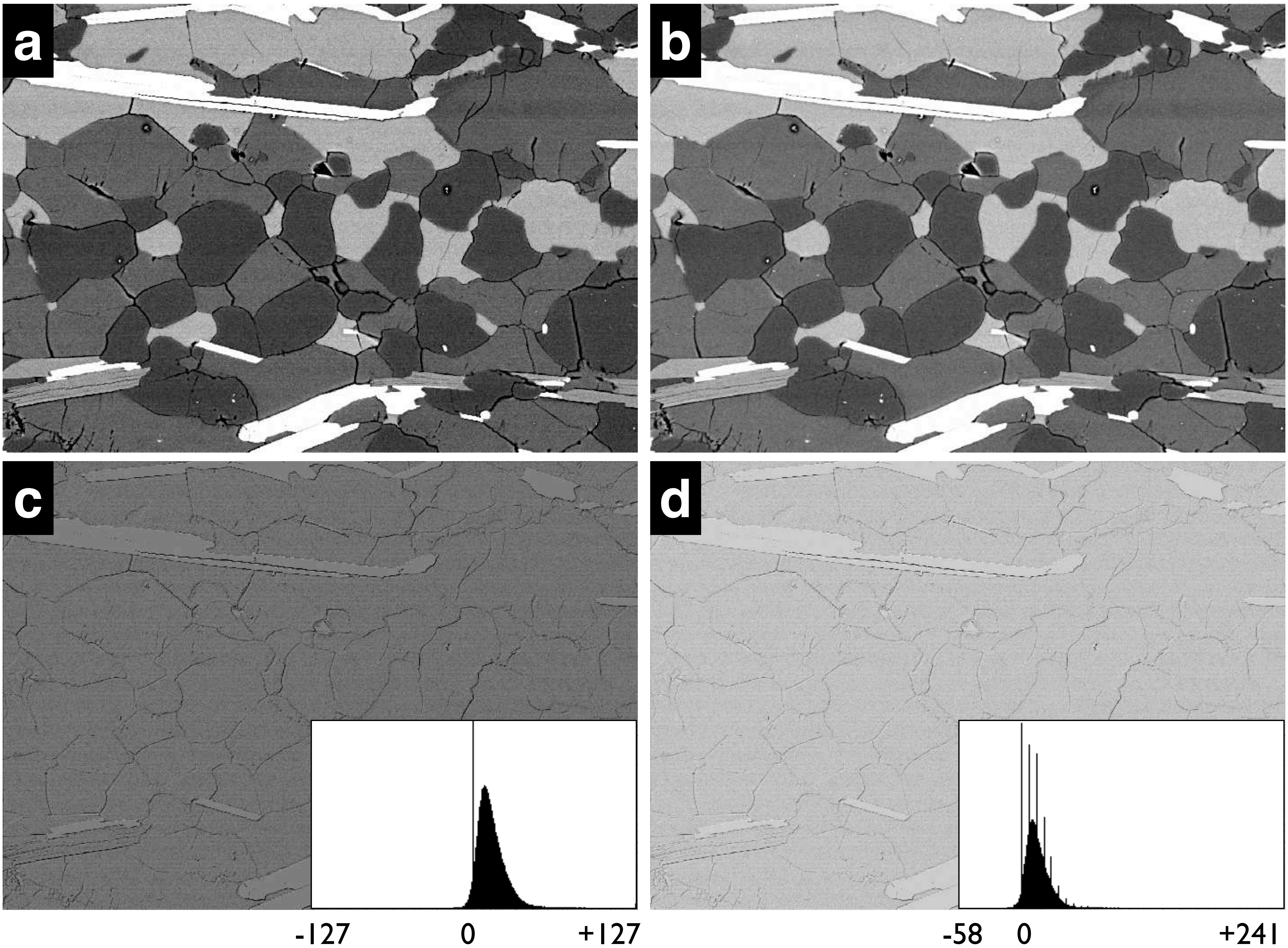

Figure 4.1
Subtracting images: detecting differences.
(a) SEM micrograph, re-sized using bicubic interpolation;
(b) SEM micrograph, re-sized using nearest neighbor interpolation;
(c) difference image (a) - (b), Grayscale result, histogram of difference image;
(d) difference image (a) - (b), Real result, histogram of difference image.

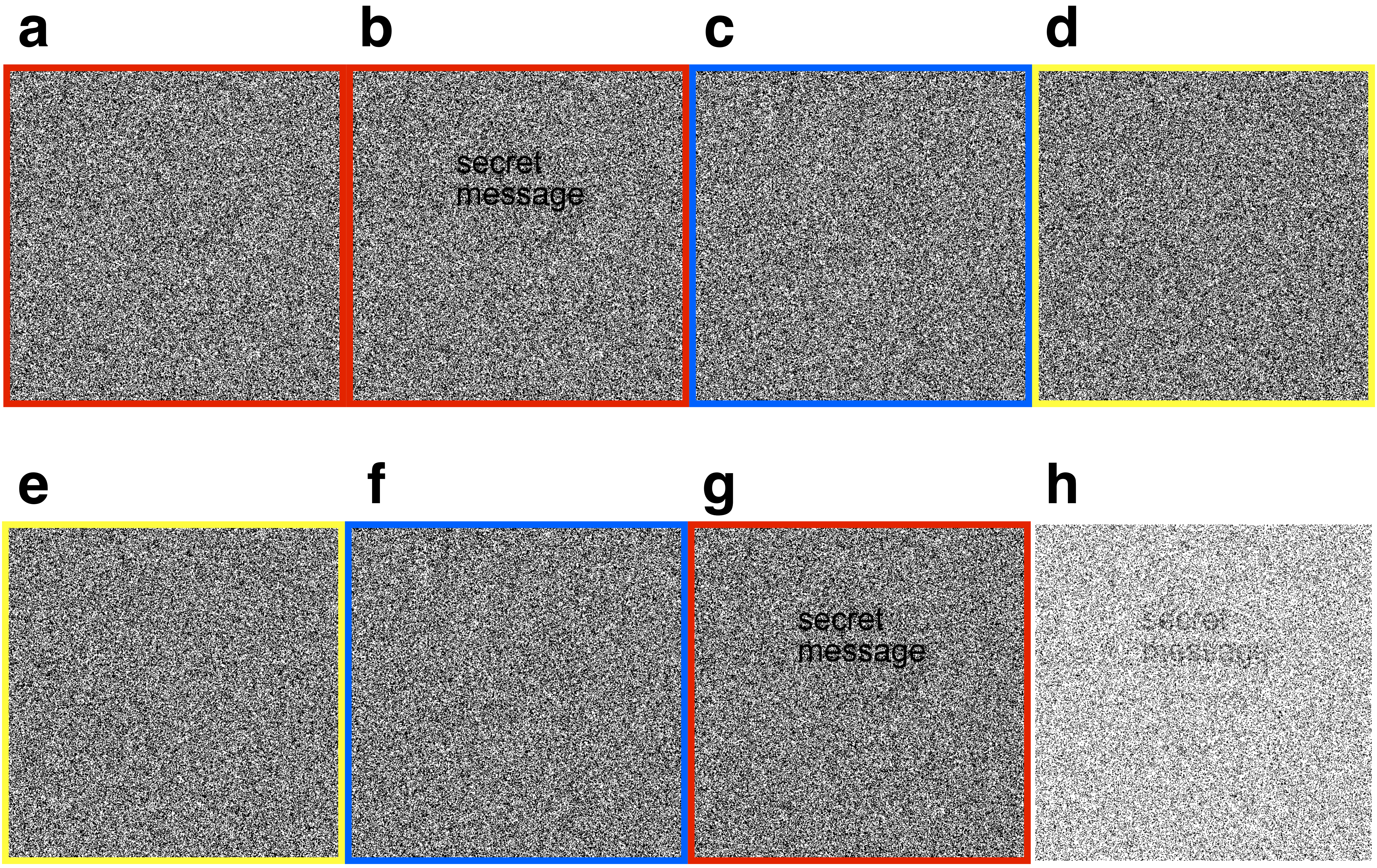

Figure 4.2
Combining images: hiding information in random noise.
Coding process:
(a) image 1 of random noise; (b) text written on image 1; (c) image 2 of random noise (≠ image 1); (d) XOR addition of (b) and (c), result = random noise.
Decoding process:
(e) same as (d) = secret message; (f) same as (c) = key; (g) XOR addition of (e) and (f); (h) subtraction: (e) — (f).

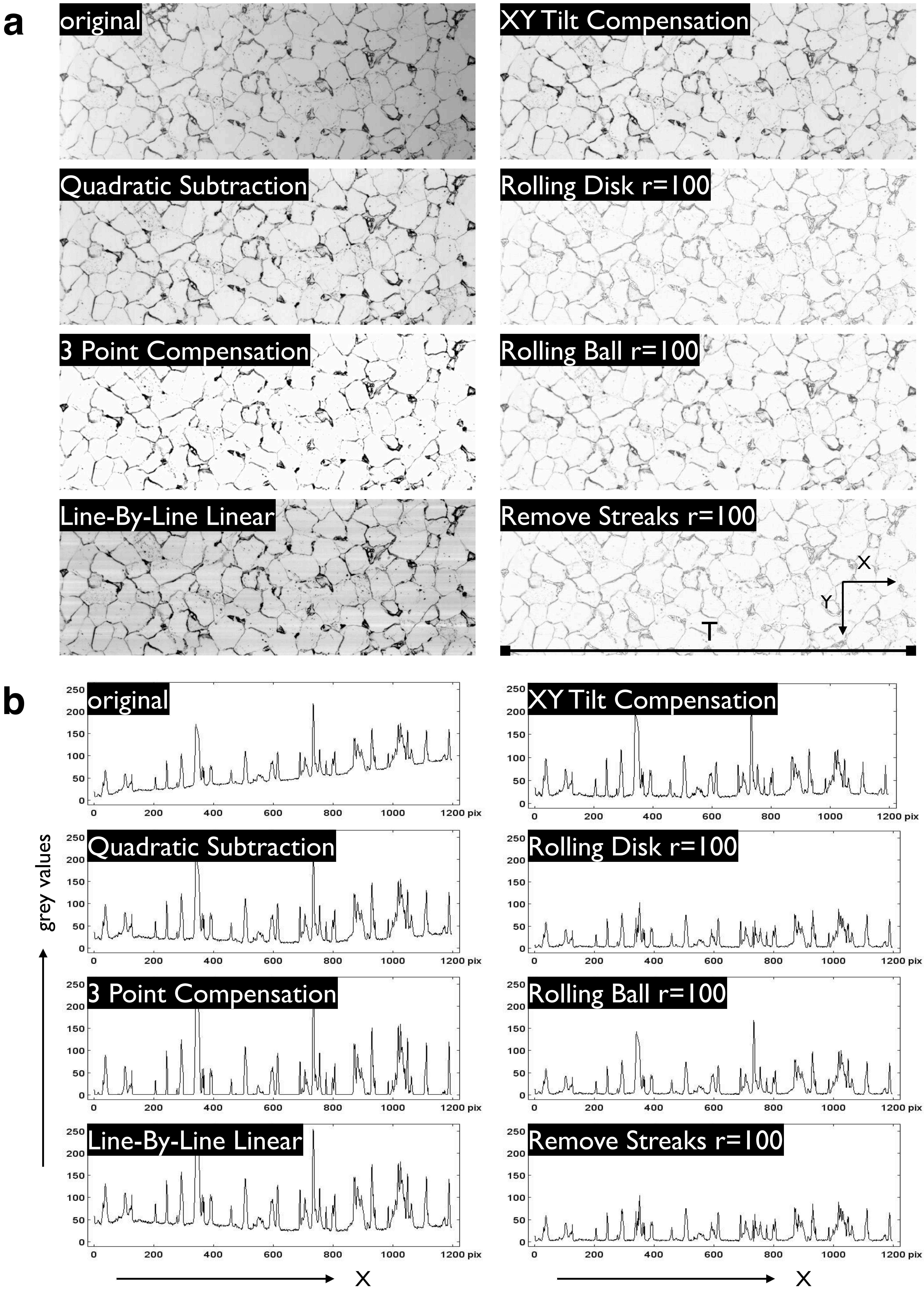

Figure 4.3
Background correction for oblique lighting.
(a) Corrected images with T marking trace of profile, the profile is at the same position in each image;
(b) gray value profiles along T.

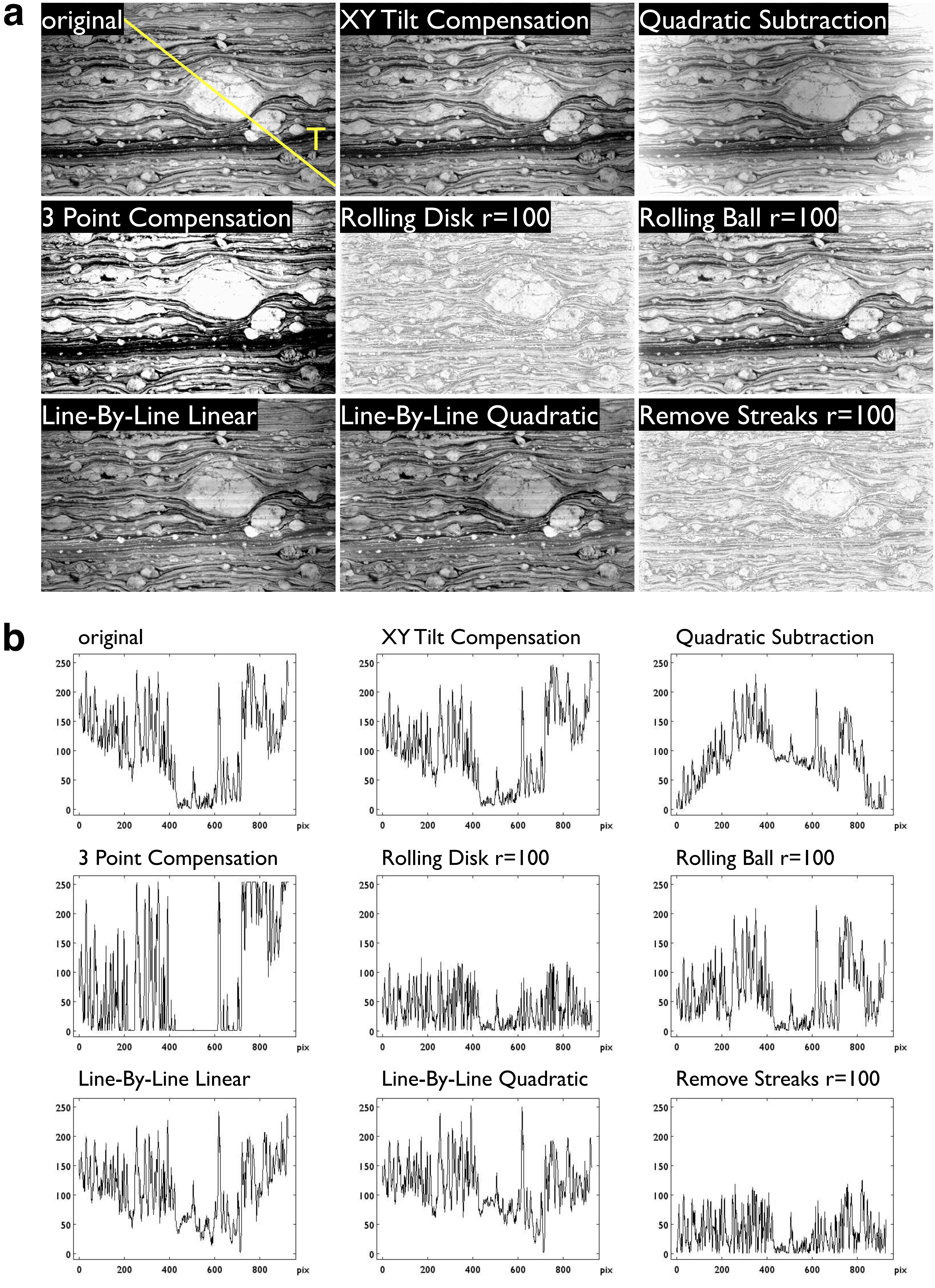

Figure 4.4
Background correction for central lighting.
(a) Corrected images with T marking trace of profiles, the profile is at the same position in each image;
(b) gray value profiles along T.

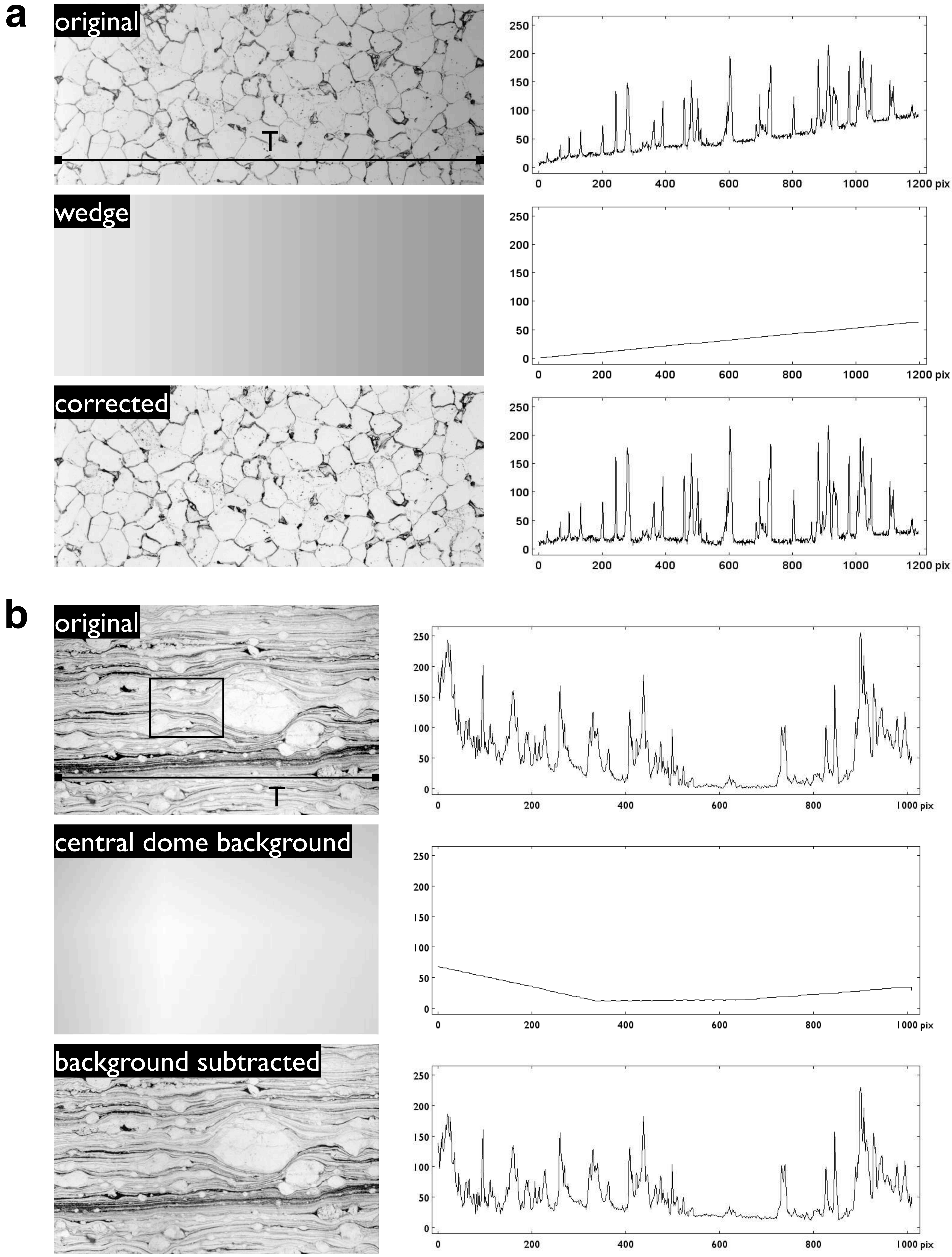

Figure 4.5
Background correction using model background.
Left: images with T marking trace of profiles; right: gray value profiles along T.
(a) A horizontal wedge is used to simulate oblique lighting from left;
(b) a central dome is used to simulate vignetting; a frame indicates the area that is used as the representative bright spot.

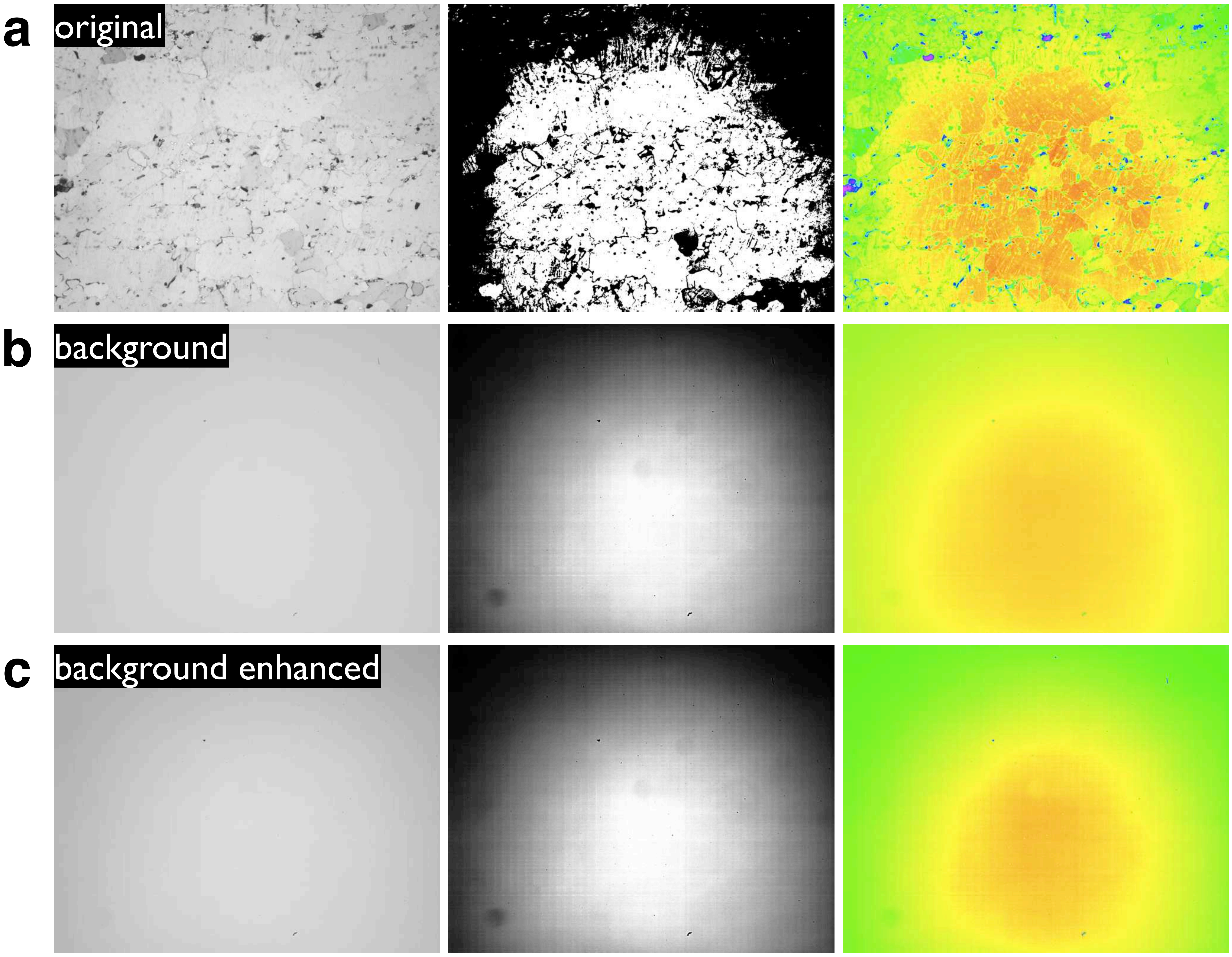

Figure 4.6
Preparing the background image.
(a) Original image, shown with a threshold and the Spectrum LUT;
(b) original background, shown with equalized histogram and the Spectrum LUT;
(c) enhanced background, shown with equalized histogram and the Spectrum LUT.

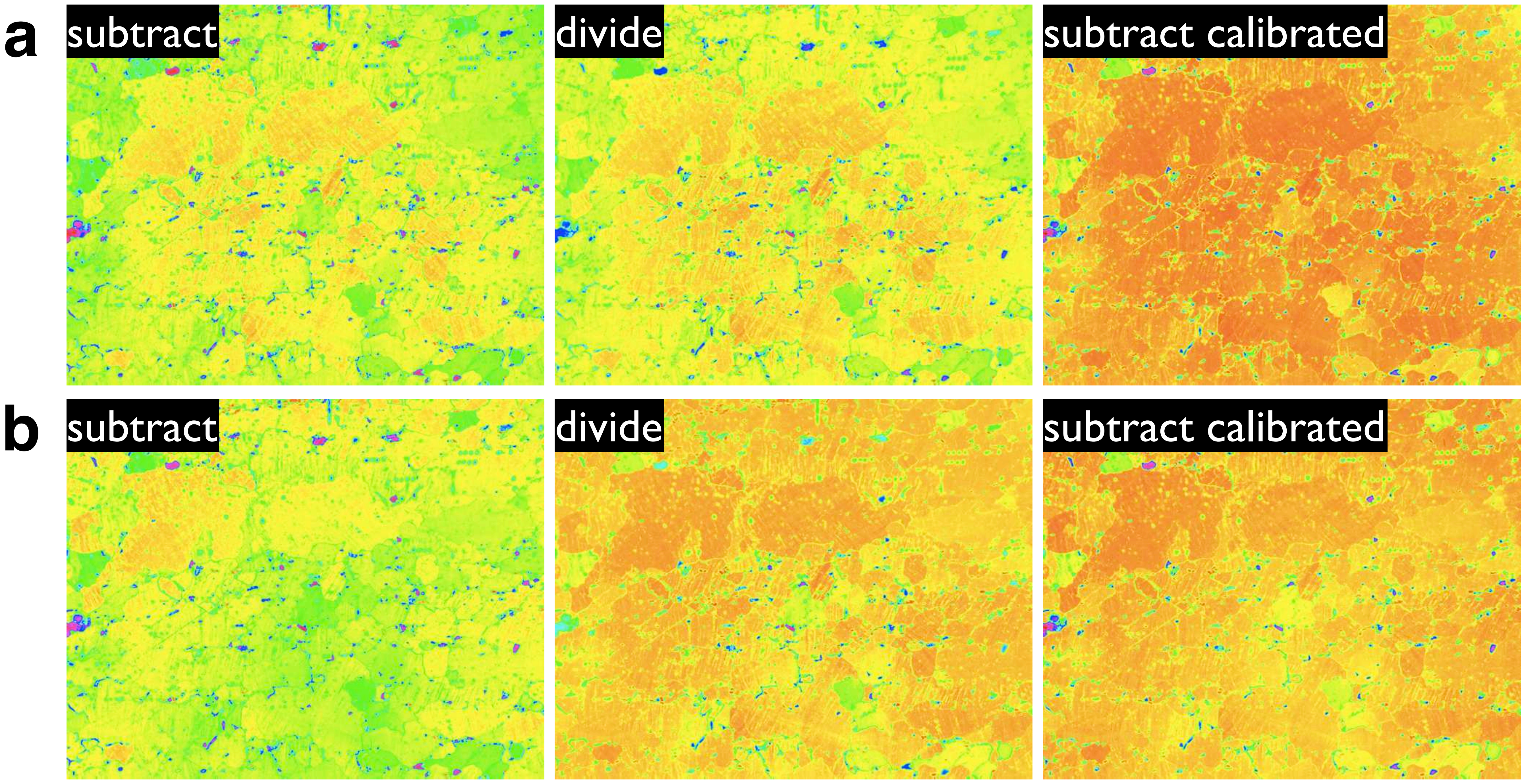

Figure 4.7
Background correction using true background images.
(a) Original background (Figure 4.6.b) is used;
(b) enhanced background (Figure 4.6.c) is used.
Three methods are used:
subtract: the background is subtracted;
divide: the original is divided by the background;
subtract calibrated: the linearized background is subtracted from the linearized original.
All images are shown with Spectrum LUT.

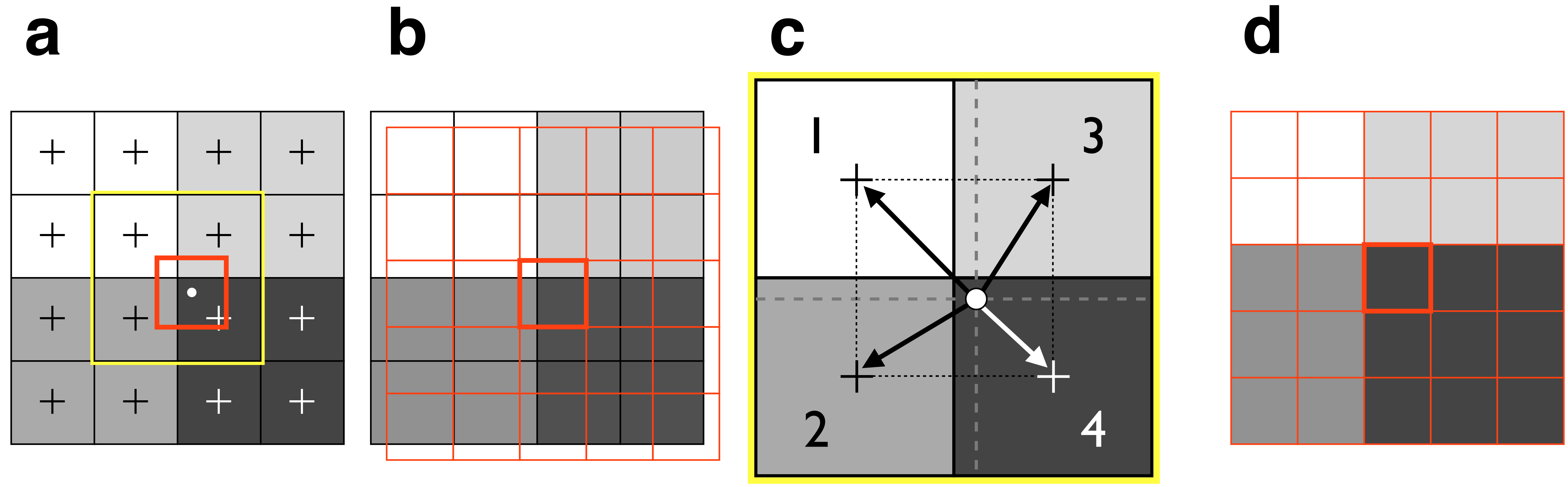

Figure 4.8
Re-sizing an image by nearest neighbor (NN) interpolation.
(a) magnified original image; one back-transformed pixel is highlighted (red); 4 neighboring pixels outlined in yellow;
(b) pixel grid; back-transformed pixel grid (red), same pixel as in (a) is highlighted;
(c) 4 neighbors of back-transformed pixel in original image (see frame in (a));
(d) result of magnification (1.2 times) of original (a) using NN interpolation.

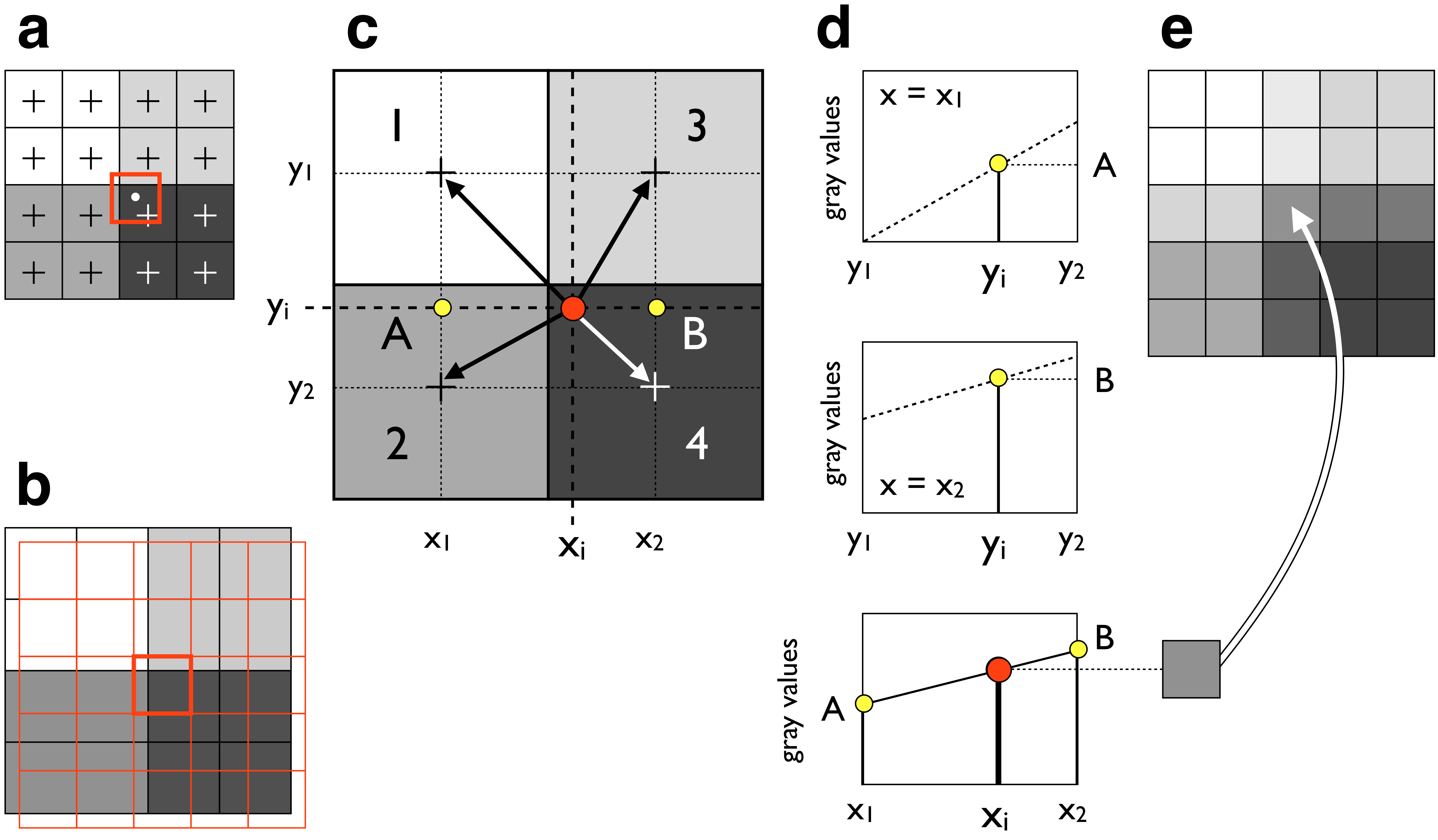

Figure 4.9
Re-sizing an image by bilinear interpolation
(a) magnified original image; one back-transformed pixel is highlighted (red);
(b) pixel grid; back-transformed pixel grid (red), same pixel as in (a) is highlighted;
(c) 4 neighbors of back-transformed pixel in original image;
(d) linear interpolations between points 1 and 2, 3 and 4, and A and B;
(e) result of re-sizing of original (a) using bilinear interpolation.

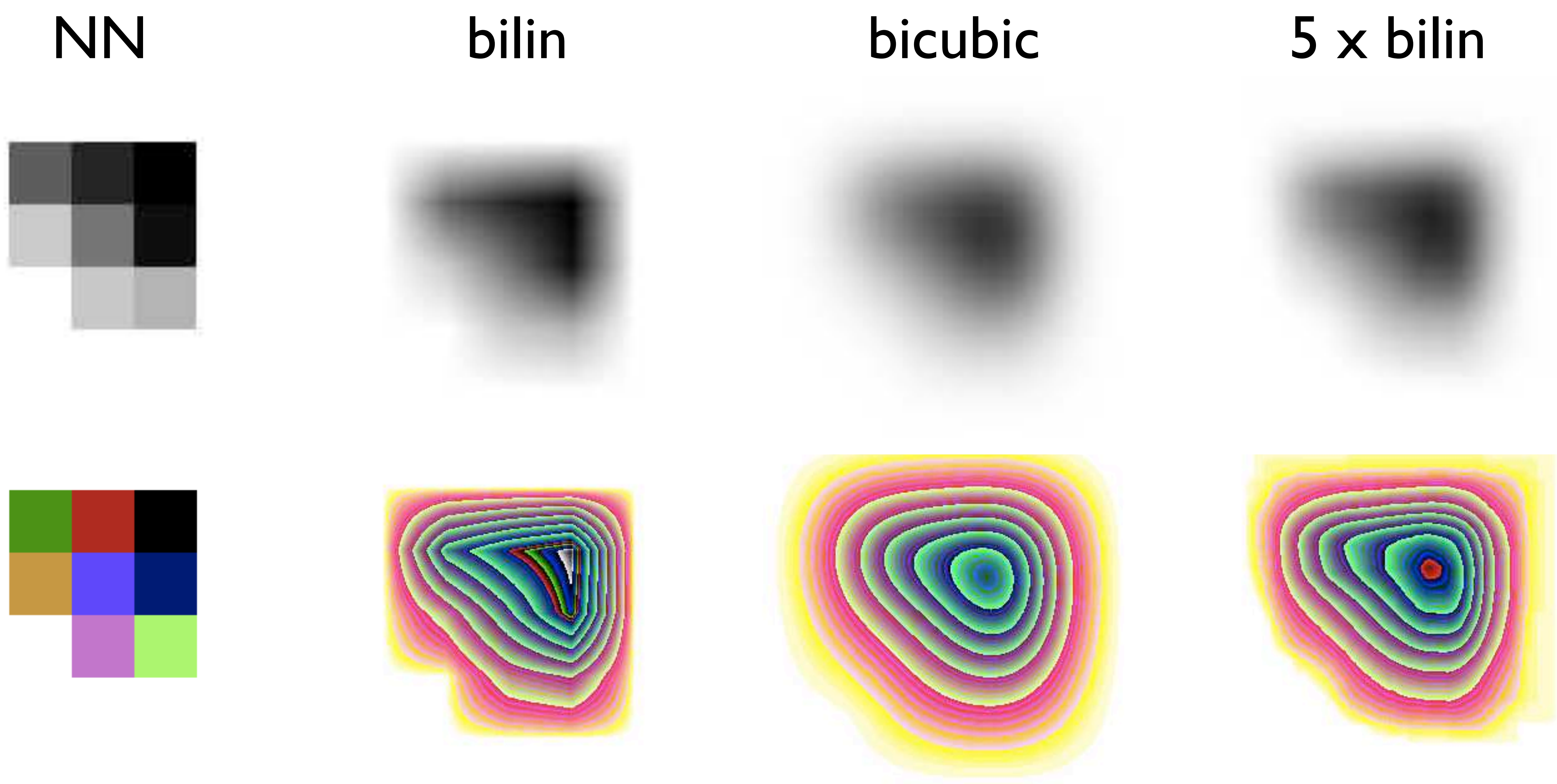

Figure 4.10
Comparison of interpolation methods.
A 3 · 3 pixel image (on a 7 · 7 background) is magnified 25 times, top: grayscale, bottom: System LUT.
NN = using nearest neighbor;
bilin = using bilinear interpolation;
bicubic = using bicubic interpolation;
5 x bilin = 5 times 1.904 x, using bilinear interpolation.

a''	880.000
g♯''	830.609
g''	783.991
f♯''	739.989
f''	698.456
e''	659.255
d♯''	622.254
d''	587.330
c♯''	554.365
c''	523.251
b'	493.883
a♯'	466.164
a'	440.000

Johann Sebastian Bach:
'Das wohltemperierte Klavier'

chromatic scale:
12 halftones per octave
1 octave = double frequency

⇒ frequency ratio between halftones:

$$\sqrt[12]{2} = 1.05946$$

analogously:

$$\sqrt[s]{M} = m$$

M	desired final magnification
s	number of steps
m	magnification at each step

Figure 4.11
Re-sizing by 'Equal Temperament'.

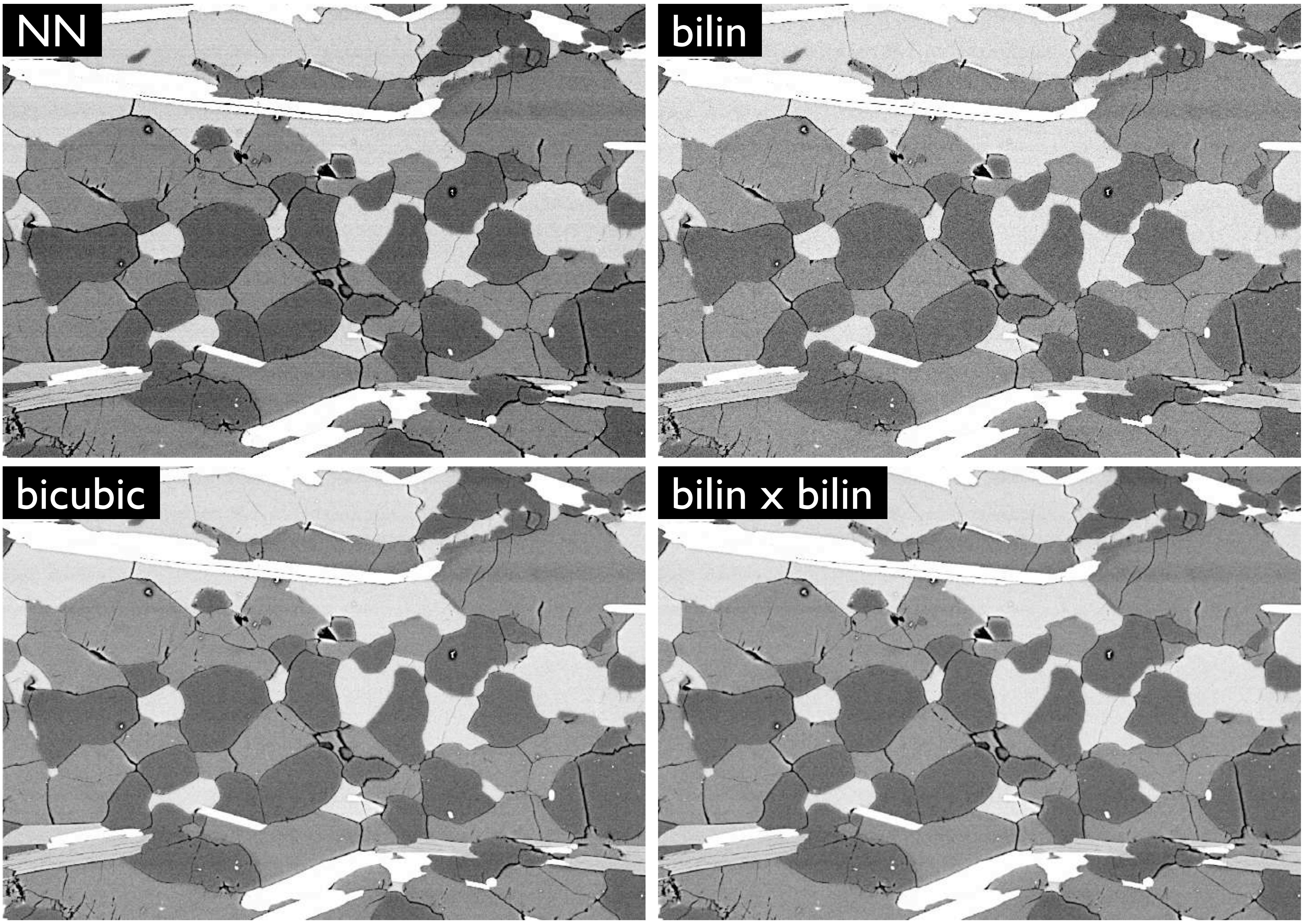

Figure 4.12
Re-sizing images using different methods of interpolation.
NN = 0.5x, using nearest neighbor;
bilin = 0.5x, using bilinear interpolation;
bicubic = 0.5x, using bicubic interpolation;
bilin x bilin = 2 times (0.707x, using bilinear interpolation).

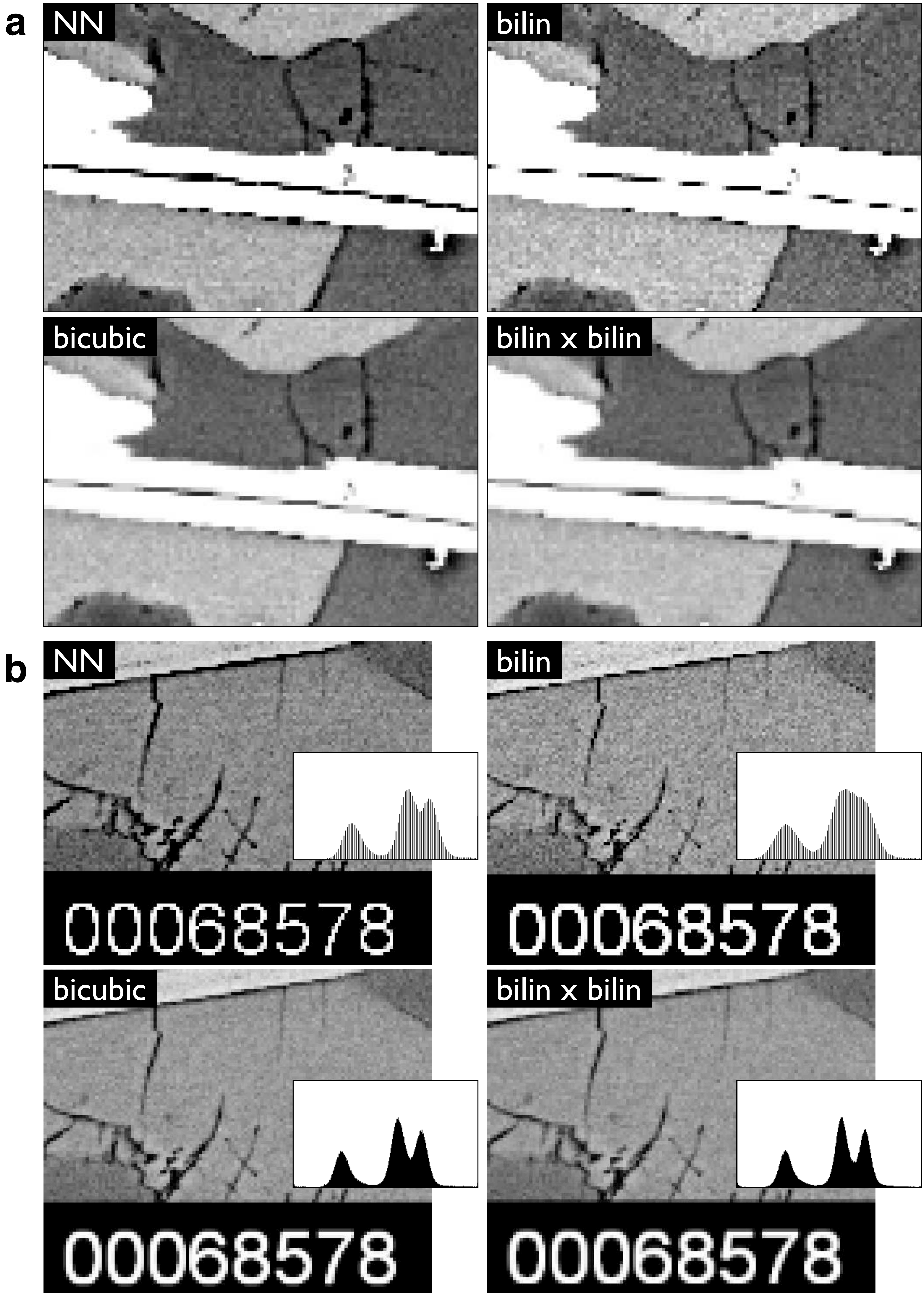

Figure 4.13
Effect of interpolation methods on noise and gray values.
(a) Detail areas after re-sizing; (b) characters and histograms after re-sizing; NN = 0.5x, using nearest neighbor; bilin = 0.5x, using bilinear interpolation; bicubic = 0.5x, using bicubic interpolation: bilin x bilin = 2 times (0.707x, using bilinear interpolation).

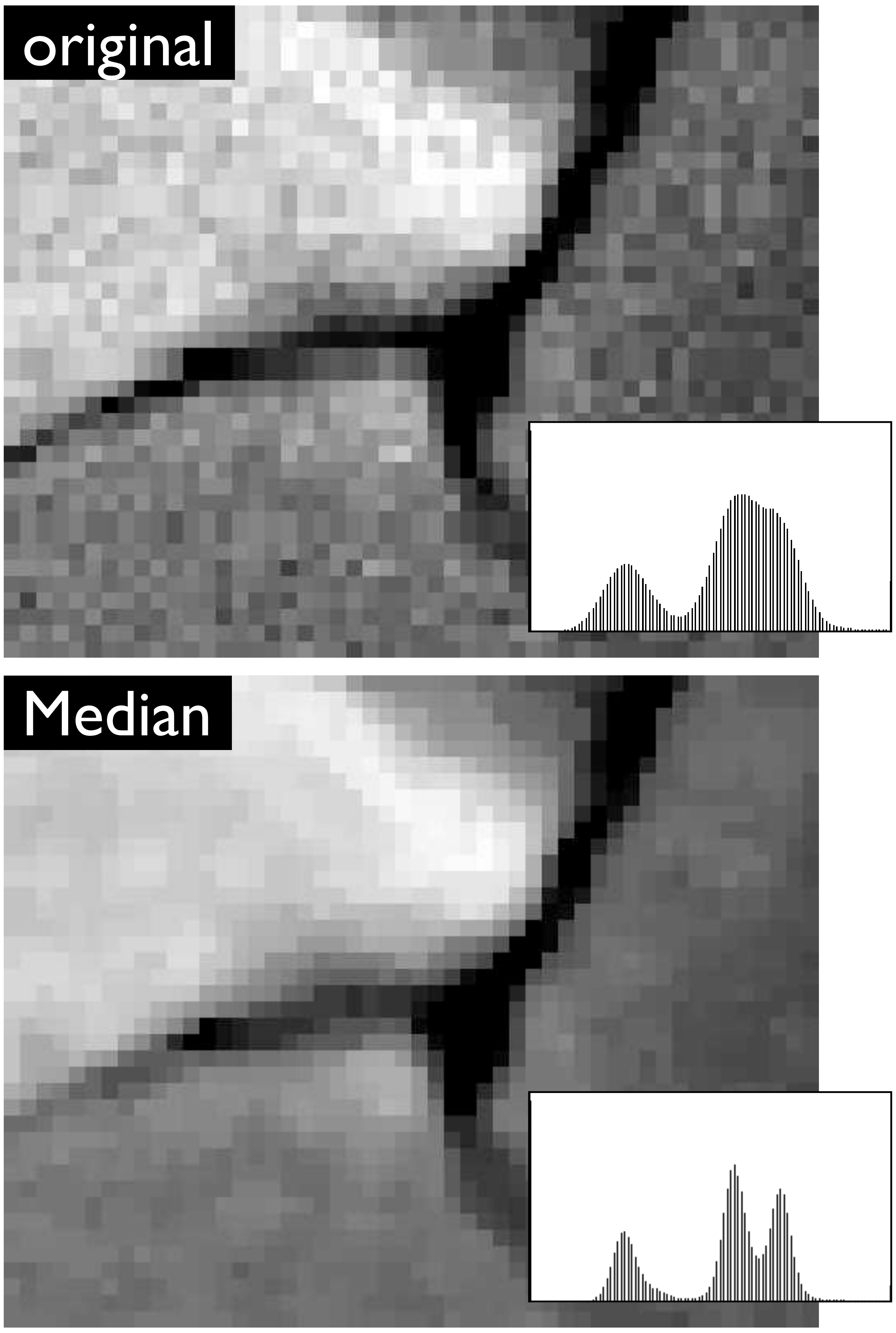

Figure 4.14
Effect of different smoothing filters.

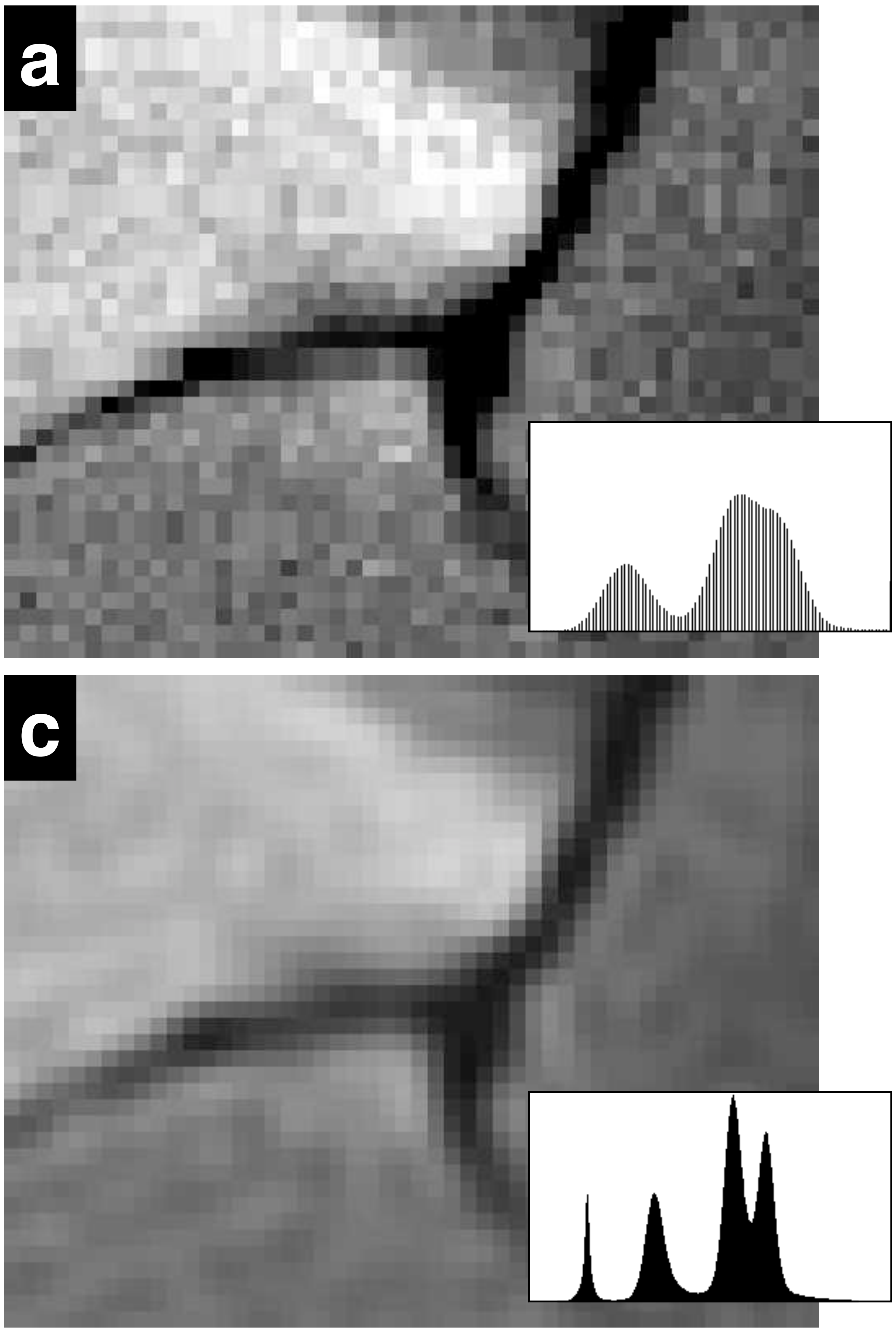

Figure 4.15
Effect of low-pass frequency filtering.
(a) Original;
(b) FFT (1424 · 1006 copied on 2048 · 2048) noise filtered radius = 768;
(c) FFT (1424 · 1006 copied on 2048 · 2048) noise filtered radius = 512;
(d) FFT (1424 · 1006 copied on 2048 · 2048) noise filtered radius = 256.

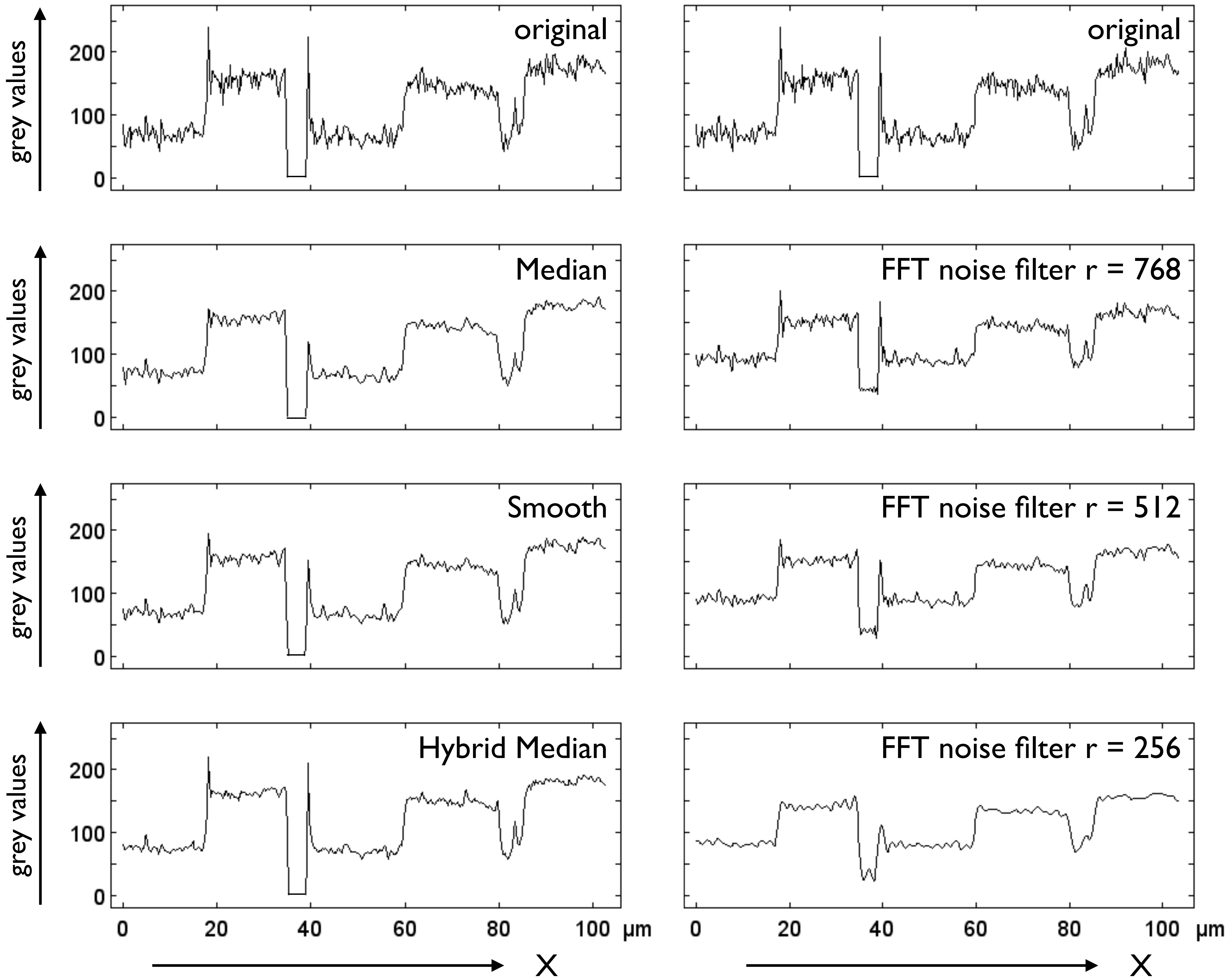

Figure 4.16
Effect of noise filtering on profile.
Trace of profile is same as shown in Figure 3.16, filter methods as in Figures 4.14 and 4.15.