Noise Reduction vs. Spatial Resolution

Uwe Artmann[a] and Dietmar Wueller[b]

[a,b]Image Engineering, Augustinusstrasse 9d, 50226 Frechen, Germany;

ABSTRACT

In modern digital still cameras, noise-reduction is a more and more important issue of signal processing, as the customers demand for higher pixel counts and for increased light sensitivity. In the last time, with pixel counts of ten or more megapixel in a compact camera, the images lack more and more of fine details and appear degraded. The standard test-methods for spatial resolution measurement fail to describe this phenomenon, because due to extensive adaptive image enhancements, the camera cannot be treated as a linear position-invariant-system. In this paper we compare established resolution test methods and present new approaches to describe the spatial frequency response of a digital still camera. A new chart is introduced which consists of nine siemens stars, a multi-modulation set of slanted edges and Gaussian white noise as camera target. Using this set, the standard methods known as SFR-Siemens and SFR-Edge are calculated together with additional information like edge-width and edge-noise. Based on the Gaussian white noise, several parameters are presented as an alternative to describe the spatial frequency response on low-contrast texture.

Keywords: Noise reduction, SFR-Siemens, SFR-Edge, MTF, Spatial resolution, Noise, resolution measurement

1. INTRODUCTION

The measurement of the spatial resolution of a digital still camera is a complicated task. As the image enhancement methods in the image processing pipeline become more complex and scene adaptive. The established methods as described in ISO 12233 (visual resolution and the measurement of the spatial frequency response *SFR* using a high contrast, slanted edge) fail to describe the response of the camera to low contrast details and texture in the scene.

A digital still camera has to be considered as a non-linear system, so it is impossible to describe the system completely using a single target. In this paper we present a chart and the corresponding software to describe the spatial frequency response of a camera more complex. We use several siemens stars, a multi-modulation edge structure and an area showing gaussian white noise.

2. NOISELAB

To learn more about denoising artifacts, the tool *NoiseLab* was written using Mathworks Matlab. This tool simulates different methods of noise reduction. The implemented methods are Median-Filtering, Wiener-Filtering, Subband-Coring and Wavelet-Coring, each with different implementations. The aim was to learn about the behavior of different methods of noise reduction in digital images. As noise reduction is part of the image pipeline in a digital still camera, and is confidential knowledge of the manufacturers and difficult to reproduce.

But we have seen that all adaptive and non-linear approaches have one point in common: The quality of the noise reduction relies basically on the ability to distinguish between information and noise in the image signal. The most artifacts are produced if image information is treated as noise (blurred and vanished fine details) or noise is treated as information (strong noise).

Further author information: (Send correspondence to Uwe Artmann)
Uwe Artmann: E-mail: u.artmann@ivent.de, Telephone: +49 2234 912141 www.image-engineering.de
Dietmar Wueller.: E-mail: d.wueller@ivent.de, Telephone: +49 2234 912141

3. NOISELAB CHART

NoiseLab is a simulation of different denoising algorithms. To compare these to real images taken with a digital still camera, the same input for camera and simulation is needed. Instead of natural scenes, a file with different test patterns has been created. The file is input for the software, a hardcopy of this file is the test chart as "input" for the camera. The chart shall be illuminated homogeneously and the camera under test shall reproduce the chart completely. The structures in the chart are chosen to measure different aspects of spatial frequency response and / or noise appearance of the camera images. The images of the NoiseLab Chart is the input for the NoiseLab Analyzer software which provides an observer independent analysis.

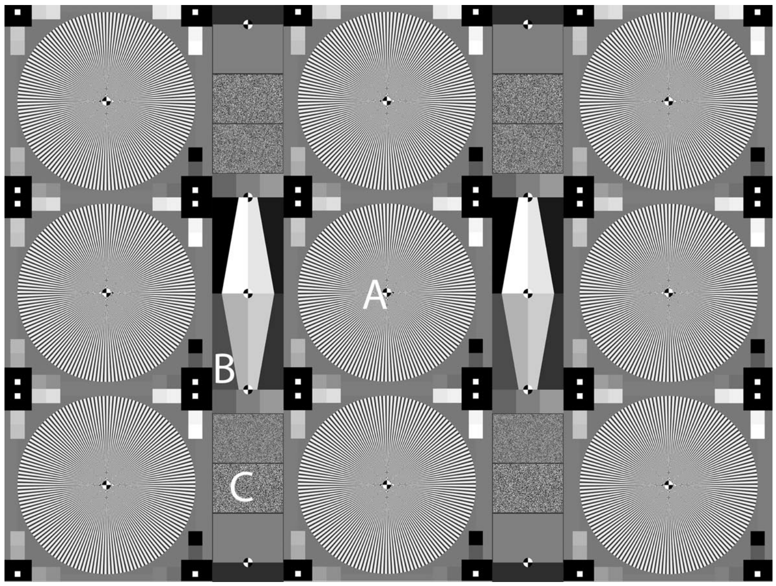

Figure 1. The NoiseLab Chart

The chart contains three main structures: (see Fig. 1)

A harmonic **Siemens stars** for SFR Siemens on 9 positions in the image, additional gray patches for linearization

B **edges** for SFR Edge, four different modulations, additional gray patches for independent linearization

C gaussian **white noise** with different noise variance (noise modulation), a gray line between patches, four flat patches without noise

Structure A is the already existing chart for the SFR Siemens method, so structures B and C can be added to the existing chart.

3.1. A -Siemens stars

The nine Siemens stars are arranged like shown in Figure 1, each star shows 144 cycles per full circle. Defining the minimum reflection as 0 and the maximum reflection as 1, the cycle is a sinus wave with a modulation of 1. In the center of each star a mark is placed, this is used for an automatic center detection in the analyzing process. 16 gray patches around the star are used for linearization of the input image. The reflection of the patches is evenly distributed between the minimum and maximum reflection.

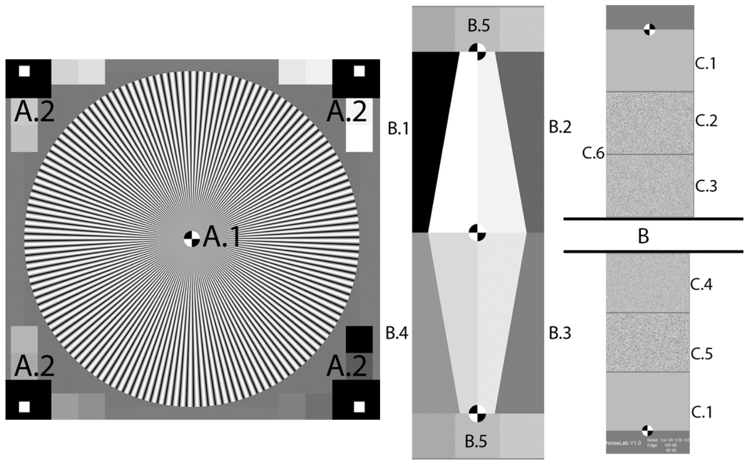

Figure 2. Details of the chart: [**A.1**] Harmonic Siemensstar, 144 cycles per circle [**A.2**] Gray patches for linearization, even distributed (reflectance) between minimum and maximum density. [**B.1**] Edge, slanted by 10°, 100% modulation [**B.2**] Edge, 80% modulation [**B.3**] Edge, 60% modulation [**B.4**] Edge, 40% modulation [**B.5**] Additional gray patches, 0.4, 0.5 and 0.6 reflectance [**C.1**] no noise, 0.5 of $D_{max} - D_{min}$ [**C.2**] gaussian white noise, $\sigma = 1/4$, mean as C.1 [**C.3**] gaussian white noise, $\sigma = 1/8$, mean as C.1 [**C.4**] gaussian white noise, $\sigma = 1/16$, mean as C.1 [**C5**] gaussian white noise, $\sigma = 1/2$, mean as C.1 [**C6**] line between patches C.2 and C.3, mean as C.1

3.2. B - Edges

The four different edges are used for the SFR-Edge algorithm. B.1 to B.4 are slanted by 10°, showing a modulation from 100% to 40%. On top and bottom, three additional gray patches have been added. Together with the edges, ten different areas with a reflection from zero (minimum reflection) to one (maximum reflection) can be read out and be used for a linearization. The patches from structure A are not used to be more flexible and to use some parts of the chart independently.

3.3. C - White Noise

Structure C consists of five different patches that show a gaussian white noise. The noise was created using Matlab and is realized that way, that the printer resolution does not limit the frequency spectrum of the noise, but it is still high enough to be able to measure cameras with up to idealized 14 Megapixel. Different variances of the noise have been implemented as well as two patches without noise, which have the same mean value as the noise patches. This is half of $D_{max} - D_{min}$. Between the noise patches, a small line with no noise is implemented.

4. NOISELAB ANALYZER

The NoiseLab-Analyzer is a software tool written using Mathworks Matlab. The software analyzes image files that show the NoiseLab-Chart which can either be created using NoiseLab or by taking an image with the camera under test. Several measurements are performed on the structures providing different information about the spatial frequency response of the camera and additional noise descriptions.

4.1. SFR-Siemens

This method was presented by Loebich, Wueller, Jaeger and Klingen[9] and proofed its reliability in several hundred camera and lens tests. It uses a siemens star with a harmonic function depending on the angle φ, taking the center of the star as the base of the angle. It provides a Modulation Transfer Function MTF which describes the loss of modulation depending on the spatial frequency $f_{spatial}$(see Eq.(1) and (2)).

$$MTF(f_{spatial}) = \frac{Modulation_{image}(f_{spatial})}{Modulation_{target}(f_{spatial})} \tag{1}$$

$$Modulation = \frac{I_{max} - I_{min}}{I_{max} + I_{min}} = \frac{a+b-(a-b)}{a+b+(a-b)} = \frac{2b}{2a} = \frac{b}{a} \quad (2)$$

It is part of the chart production to take care that the $Modulation_{target}(f_{spatial})$ is 1 for all frequencies used for the measurement. The intensity I in the ideal digital image as a reproduction of the chart is

$$I(\varphi) = a + b\cos\left(\frac{2\pi}{g}(\varphi - \varphi_0)\right) \quad (3)$$

with angle φ (4), period lenght g see (5), mean value a, amplitude b and phase φ_0.

$$\varphi = \arctan\left(\frac{x}{y}\right) \text{ with } x, y = \text{projected coordinates in image} \quad (4)$$

$$g = \frac{circumference\ [pixel]}{number\ of\ periods} = \frac{2\pi r}{n_p}[pixel] \quad (5)$$

The image that shows the siemens star is read in and linearized, using the gray patches arranged around the star. The aim is to obtain the modulation depending on the spatial frequency which is defined by the radius r. The image coordinates are projected to the star coordinates, which sets the center of the star to x=0 and y=0. The star is subdivided into 24 segments to provide information on orientation specific differences. The modulation is obtained depending on three variables: radius r and the starting and ending angle φ_{start} and φ_{end}

The base for the calculation is the function $I(r, \varphi)$ which is directly read out of the image data. The pixels that are located best to the ideal circle with radius r are used to build up the 1D data-array.

After the reading for one radius r the row vector $I(\varphi)$ and the row vector φ are known. Using (3) the unknown variables are the phase φ_0 and a and b. To get the phase out of the calculation the approximation (6) is used instead of (3).

$$I(\varphi) = a + b_1 \sin\left(\frac{2\pi}{g}(\varphi)\right) + b_2\cos\left(\frac{2\pi}{g}(\varphi)\right) \quad (6)$$

The mean a and the amplitude b_1 and b_2 are calculated using the least square error fit method (left division in Mathworks Matlab) which fits an idealized harmonic function to the obtained image intensity data. With the geometric mean of b_1 and b_2 the modulation is calculated as mentioned in (2). So the output of the calculation is the function $M(f, \varphi_{start}, \varphi_{end})^*$. So the output are up to 24 MTFs for each of the 9 stars in the image. Using this data, different parameter can be calculated like MTF50, or the limiting resolution, which is here MTF10.

4.2. SFR-Edge

The SFR-Edge algorithm is described in ISO12233. The new approach is to use a multi-modulation edge target, starting from 100% to 40% Modulation. The assumption is, that the denoising algorithms have to detect edges to distinguish between information and noise in the image signal. Using the different modulations, it can be checked if the edges are treated differently. The image is read in and linearized using the areas forming the edge and the adjacent patches above and below the edges. The image regions just showing the edge are determined using the positions of the marks and used as input for the algorithm. The algorithm is written based on the information in the ISO standard[6] and the documentation to the public software *sfrmat* by Peter Burns.[10]

First step is to localize the edge in each row. Using this data, the offset and the slope of the edge in the image is calculated. The edge description is used to calculate an over-sampled pixel row. This is done by a binning process, placing each pixel of the image into a bin which describes a certain distance to the fitted edge. So the two-dimensional position of each pixel with column x and row y becomes a one-dimensional description with its distance to the edge.

The over-sampled description of the edge is called the edge spread function *ESF*. The first derivative of the ESF is the line spread function *LSF*[†]. The SFR-Edge is the Fourier transform of the LSF. Before the transformation, the data is windowed to avoid leakage.

[*] Modulation depending on spatial frequency f, starting angle φ_{start} and ending angle φ_{end}

[†] The LSF can be imagined as a 1-D representative of the point-spread function PSF

4.3. Edge Profile - Intensity and Standard Deviation

The edge profile is obtained while calculating the SFR-Edge. The intensity edge profile is the ESF as mentioned in section 4.2.The deviation profile of the edge is calculated by an additional step in the SFR-Edge calculation. It represents the standard deviation of digital values in the pixel columns, paralleled to the edge. This makes it possible to measure the noise along an edge, which is interesting to measure, because most denoising algorithms keep a certain distance to an edge to perform the denoising.

The edge profile is described in two ways. The first one is to provide the edge profile based on its intensity, which is the ESF. Using this data, the edge-width as a 10% to 90% rise is calculated and accompanied by different parameter to describe the undershoot and overshoot as sharpening artifacts.

While the intensity profile uses the mean value of all pixels with a certain distance to the fitted edge, the second description uses the standard deviation between all pixels with a defined distance to the edge. So an increase of the noise in presence of an edge can be visualized and measured.

4.4. SFR-Noise

The SFR-Noise is an experimental approach of measuring the SFR using a gaussian white noise as input. The first, meanwhile dismissed approach was to obtain the SFR just by calculating the noise power spectrum in the image of the white noise. White noise means that the noise spectrum equals one for all spatial frequencies. So by knowing the spectrum of the transfered noise, the SFR is known. This method works well for linear systems, so if the blurring is just induced by the lens. Tests on images taken with digital still cameras showed problems. A good SFR result could be the result of a slight low-pass filtering or of a single narrow peak in the image.

The now used method measures the correlation of the pixel values in the image. So for the region-of-interest the correlation of pixel among themselves is calculated. A gray-level-cooccurence-matrix GLCM is produced for each pixel and its neighbor in distance x, while x gets the values 1 to 20. So one gets the correlation depending on distance x. For details on GLCM see section 4.7.

The correlation vector and a symmetric copy are placed in the correlation spread function CoSF. The fast fourier transformation of the CoSF is the SFR-Noise.

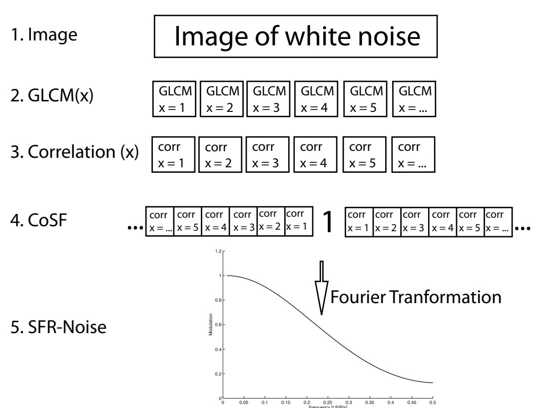

Figure 3. Calculating SFR-Noise

While testing this approach on different cameras, it became obvious that the results of SFR-Noise do not have an advantage against SFR-Siemens to describe the spatial resolution in presence of noise reduction. But it does provide useful information in the comparison of SFR-Noise obtained from image sections showing target noise or from a flat field section. The less the difference in SFR-Noise on the different patches, the less the difference between target noise and system noise, the more the images appear degraded.

4.5. Line Profile

While evaluating with different cameras, one artifact of image denoising became obvious. The stronger the noise reduction due to high noise level in the image, the more the small line between the noise patches disappears. Figure 4 shows details of images taken of the NoiseLab chart with a consumer still camera, setting the ISO speed to ISO100, ISO400 and to ISO1600.

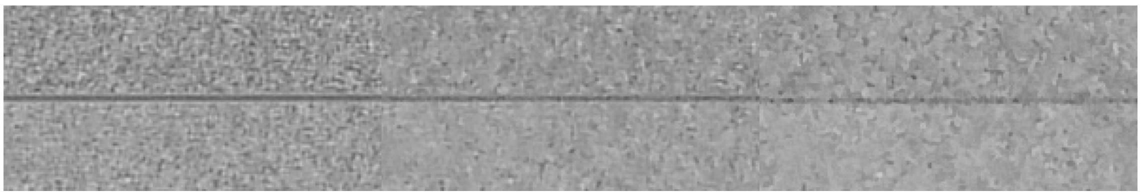

Figure 4. Line Image (detail, enlarged 2x) ISO100, ISO400, ISO1600

One can see, that with increasing ISO speed the differentiation between line and noise patch becomes more and more difficult. As the mean value of the noise and of the line should be the same (zero mean gaussian white noise) the line profile is measured using a std-filter.

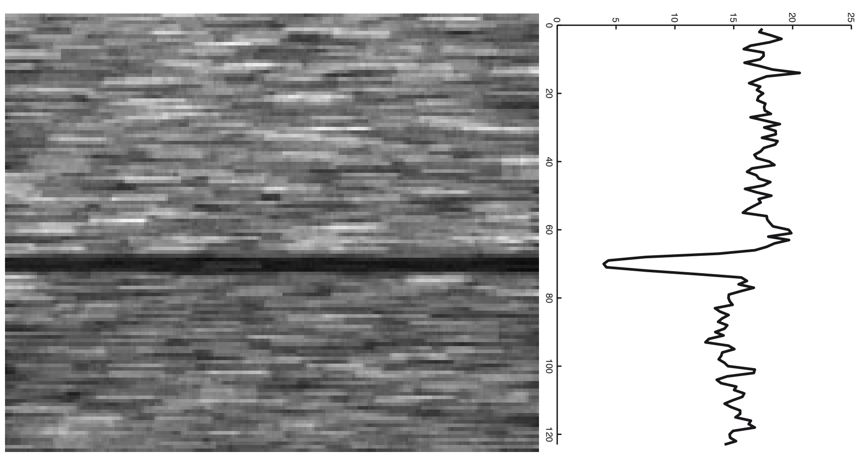

Figure 5. Image after std-filtering for Line Profile

The std-filter calculates the standard deviation in a neighborhood around each pixel. In NoiseLab-Analyzer the neighborhood is a single horizontal line, 10% in length of the full row. The neighborhood spans not the full row to minimize the influence on the result if the image is slightly tilted. The Line Profile is the mean value of image rows as illustrated in figure 5.

4.6. Histogram of Derivative

The histogram of the image parts showing the noise structure contains different information about the noise characteristics. But the histogram would change for different mean values of the noisy image signal, so in NoiseLab, the histogram of the first derivative is used. This is calculated by a convolution of the image with the kernel [-.5 .5]. The first derivative of a normal distribution also contains normal distribution and so on, so it is possible to check for the distribution in the derivative image.

The target is a gaussian white noise, so all digital values appear in the image with a probability defined by the gaussian distribution around the mean value. In the processed image, the mean value is zero, as the first derivative of a flat image is zero. The more the image is low pass filtered, the more the histogram gets a peak at its mean and the more the distributions become *leptokurtic*‡, so the probability of values close to the mean is increased.

To describe the shape of the distribution, the *excess kurtosis* is calculated. The value becomes 0 for a normal distribution and is increased for leptokurtic distributions. The kurtosis is calculated as the fourth moment devided by the square of the second moment of the distribution. The second moment is the variance.

$$kurt = \frac{m_4}{m_2^2} - 3 = \frac{m_4}{\sigma^4} - 3 = \left(\frac{1}{n} \sum_{i=1}^{n} \left(\frac{x_i - \mu}{\sigma} \right)^4 \right) - 3 \tag{7}$$

4.7. Noise GLCM

> "A statistical method of examining texture that considers the spatial relationship of pixels is the gray-level co-occurrence matrix (GLCM), also known as the gray-level spatial dependence matrix."[11]

The GLCM is a technique from texture analysis in images. The matrix has the size of n^2 where n is the number of possible gray values in a digital image. So for an 8-bit image, the GLCM has the size 256×256. The matrix is created by counting how often a certain combination of two pixel values with distance x occurs in the analyzed image. So for example the matrix entry $GLCM(100, 120) = 250$ with distance 1 means, that 250 pixel have the value 100 and the pixel on the right next to it has the value 120.

That part of the image that shows the noise structure is used to calculate a GLCM. To make the results more comparable, the matrix values are normalized to the total number of pixel in the image. A value of 0.25 in the NoiseLab-Analyzer GLCM means that 0.25% of all pixels have the defined relationship to the neighbor pixel. The more pixel have the same value than their related neighbor pixel, the higher the diagonal coefficients in the GLCM, the less correlated the noise is, the lower are the single coefficients in the matrix and the lower are the diagonal entries.

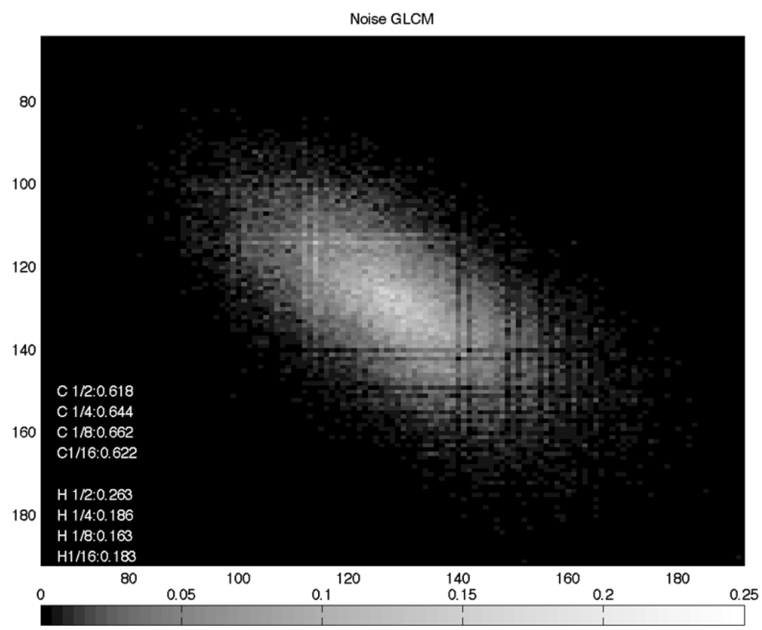

Figure 6. Noise GLCM

‡"A frequency function with coefficient of kurtosis greater than zero is said to be leptokurtic. It is more peaked about the mode than the normal distribution".[12]

Correlation is a measure for the interdependence of two variable quantities.[2] In this case it measures the likeliness, that a pixel and its neighbor have the same value. A flat field would mean that the pixel are perfectly correlated, the value would be one.

$$Correlation = \sum_{i,j=0}^{N-1} GLCM(i,j) \frac{(i-\mu)(j-\mu)}{\sigma^2} \tag{8}$$

The homogeneity measures the closeness of the distribution to the diagonal. So the more values are placed on the diagonal, the higher the homogeneity gets. The value ranges from zero to one.

$$Homogeneity = \sum_{i,j=0}^{N-1} \frac{GLCM(i,j)}{1+|i-j|} \tag{9}$$

The GLCM and the two parameter Correlation and Homogeneity descibe the appearance of the noise in the image, which is a combination of target noise (chart) and camera noise.

5. RESULTS

The NoiseLab-Analyzer has been used to test several digital cameras. While keeping the illumination and the camera setup constant, the ISO speed has been set to 100, 400 and 1600 if available.

It could be shown, that the edge-analysis method is an easy approach that provides useful information about sharpening and the edge-noise. The derived SFR-Edge depends on the edge modulation. A non-adaptive noise reduction method would lead to same results for all different contrasts. The comparison of the SFR-Edge for different edge modulations and/or ISO speed setting of the camera provides useful informations about the denoising method. The approach to describe the noise depending on the distance to the edge is useful and can extend the noise analysis in camera tests.

SFR-Siemens is a reliable method to test for spatial resolution. It is less influenced by image enhancements and is useful to provide the systems MTF. But it fails to describe the loss of texture in images for camera tests. Some cameras show strong denoising artifacts while still getting good SFR-Siemens and SFR-Edge results.

The SFR-Noise approach could not proof its benefit in the camera tests, there is no advantage against the SFR-Siemens method. The differences in SFR-Noise of gray patches and of patches with target noise showed a coherence with the visual impression of the images.

The degradation of lines in the image is one aspect of noise-reduction in digital images. The line profile is a good tool to describe this phenomenon.

The kurtosis of the image-noise, where image-noise is a combination of target-noise and camera-noise, is a good measurement for describing the non-linearity of a digital camera. It could be seen that the higher the kurtosis, the more the images appear to be degraded and show a lack of fine detail.

The GLCM as a texture analysis tool is very useful to visualize the influence of noise reduction on the image. There should be more research on the correlation between visual impression and GLCM results.

6. CONCLUSION

The SFR-Siemens or SFR-Edge methods do not fully characterize a camera system. The loss of texture as experienced in modern digital still cameras can not fully be described using these methods. We propose the additional use of other methods to test for the spatial frequency content. It could be shown, that the kurtosis of the image signal showing target noise is a good measure of the non-linearity of the camera system. So the higher the kurtosis the less the results of SFR-Siemens or SFR-Edge represent the loss of texture.

Additional informations like SFR-Noise, the Line-Profile and a GLCM provide more complex descriptions of the camera system and can improve the measurement.

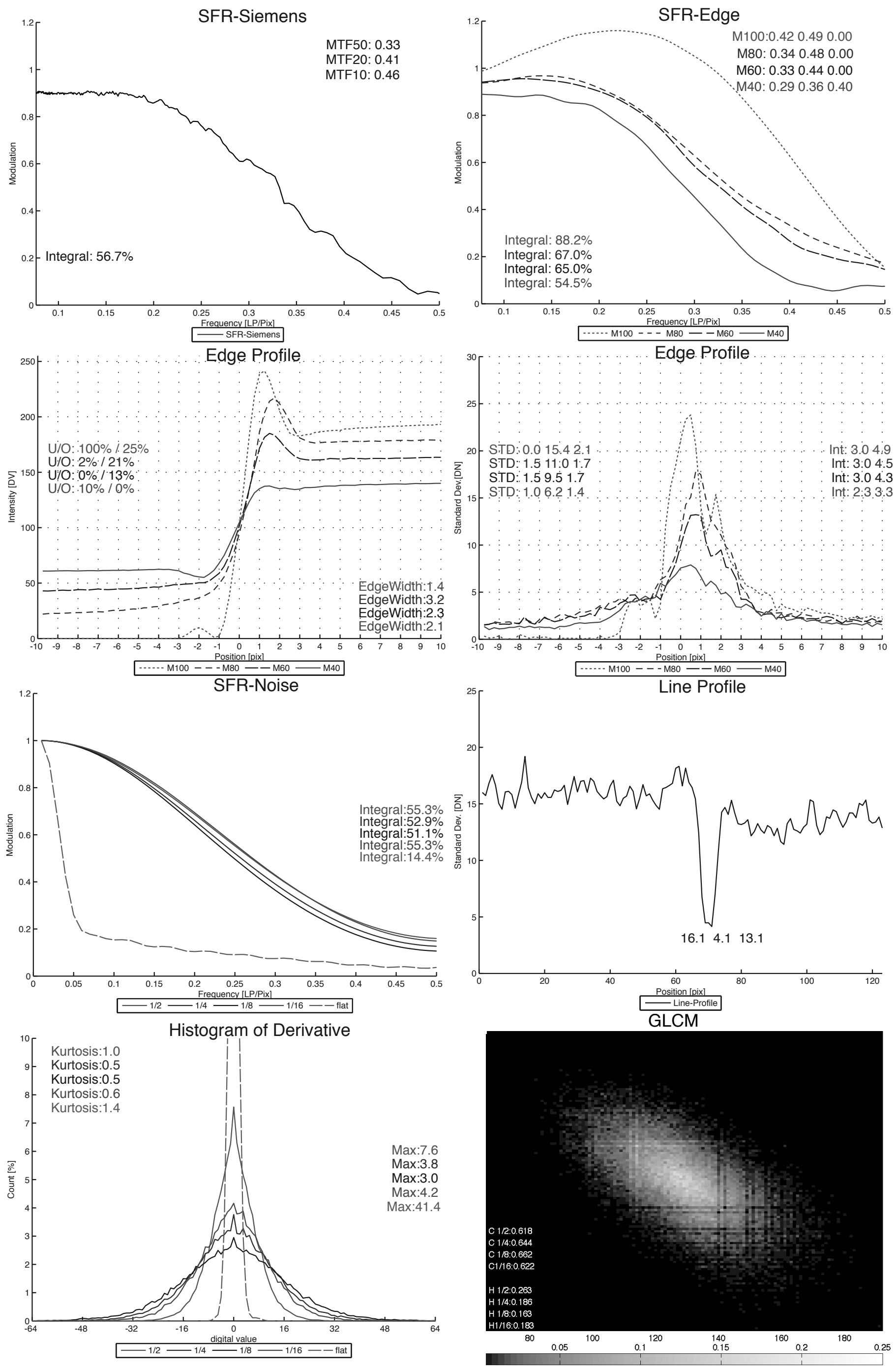

Figure 7. Graphical Results of a consumer compact camera (ISO100)

REFERENCES

1. Uwe Artmann "Noise Reduction vs. Spatial Resolution", Diploma thesis at University of Applied Science Cologne Germany, downloadable via www.image-enginering.de
2. Apple Computer Inc. "New Oxford American Dictionary, 2nd Edition"
3. Gonzales, Woods, "Digital Image Processing", 2nd International Edition, Prentice Hall, 2002
4. Prof. Dr. Kunz, University of Applied Sciences Cologne, "Digitale Bildverarbeitung", scripts to lectures
5. Loebich, Wueller, "Three Years of Practical Experience in Using ISO Standards for Testing Digital Cameras", IS&T/PICS Final Program and Proceedings,2001, pp. 257-261.
6. International Organization of Standardization,"ISO12233:2000 Resolution measurements"
7. Gonzales, Woods, "Digital Image Processing using Matlab", Prentice Hall, 2004
8. Anke Neumann, "Verfahren zur Aufloesungsmessung digitaler Kameras", Diploma thesis at University of applied Sciences Cologne, , downloadable via www.image-enginering.de
9. Loebich,Wueller,Klingen,Jaeger, "Digital Camera Resolution Measurement Using Sinusoidal Siemens Stars", Electronic Imaging Conference 2007
10. Peter Burns, "sfrmat 2.0 User's Guide", download via www.i3a.org
11. Mathworks, Documentation Image Processing Toolbox, "Analyzing the Texture of an Image", Version 6.0
12. University of Cambridge "http://thesaurus.maths.org/" Entry "Leptokurtic" 17.11.2007
13. Steven W. Smith, "Digital Signal Processing, A Practical Guide for Engineers and Scientists", Newnes, 2003
14. Bernd Jaehne, "Digitale Bildverarbeitung", Springer Verlag, 2005
15. Bruno Klingen, "Fouriertransformation fr Ingenieur- und Naturwissenschaften", Springer Verlag, 2001
16. Vinay K.Ingle, John G. Proakis, "Digital Signal Processing using MATLAB". Thomson, 2007