TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms

N. Asuni[1] and A. Giachetti[2]

[1]Tecnick.com LTD, UK
[2]Dep. Computer Science, University of Verona, Italy

Abstract
We present the TESTIMAGES archive, a huge and free collection of sample images designed for analysis and quality assessment of different kinds of displays (i.e. monitors, televisions and digital cinema projectors) and image processing tecnhiques. The archive includes more than 2 million images originally acquired and divided in four different categories: SAMPLING and SAMPLING_PATTERNS (aimed at testing resampling algorithms), COLOR (aimed at testing color rendering on different displays) and PATTERNS (aimed at testing the rendering of standard geometrical patterns). The archive is currently online as a SourceForge project and, even if not yet publicized in the scientific community, it has already been used in different contexts and cited in scientific publications. We plan to extend the archive including datasets for other kinds of specific analyses.

1. Introduction

A huge amount of research and technical work is performed in the academic and industrial domain on image processing, rendering and perception. Topics like display optimization and human perception of images [MWDG13], image upscaling and downsampling [NM14,KSP13] are extremely hot for both research and development of popular products like TV sets, projectors, smartphones, tablets, gaming devices, head mounted displays, digital cinema projectors, etc.

A critical and time-consuming activity in the development of related study is the creation of test images to evaluate quantitatively display features, perform psychovisual studies and to test algorithms for basic processing methods (e.g. resampling, denoising), evaluate shading effects or texture compression.

The availability of public datasets with images with different features and resolutions is therefore particularly important to facilitate the work of researchers, allowing them also to compare results of processing tools, both visually and quantitatively on the same data.

Despite this fact, it is not easy to find free and ready to use sets of high-quality general purpose images on the Internet. Research work on image filtering and rescaling, color perception, etc. usually presents experimental results on small sets of custom example images, making it difficult to repro-duce the experiments. For example, the metrics calculated by several authors using the iNEDI image zooming algorithm [AG08] are often discordant due to the use of different and inadequate datasets.

We therefore decided to create a large selection of images that can be used for a variety of basic research and technical applications that are collected in a unique and large archive and distributed with a Creative Commons License in order to help the Computer Graphics and Image Processing community. All the images have been originally created in order to guarantee uniform and high-quality features. Peculiar features of the dataset are the inclusion of 16 bit-depth reference images and different downscaled versions of reference high-resolution images to enable more accurate comparison of zooming algorithms.

2. Image creation and file format

All images were generated using a range of custom Octave/MATLAB [GNUa] software scripts specifically written for this purpose to guarantee the precise positioning and value of every pixel. These scripts are not provided on purpose, to guarantee that every researcher is using exactly the same images without any alteration that may occur running the same script on different platforms.

The same image is generally available in 8bpp and 16bpp

(bit-per-pixel or bit-dept) in 3-Channel Red-Green-Blue (RGB) and 1-Channel Grayscale format.

Since the intensity levels for a 16bpp image are $2^{16} = 65536$, and for a 8bpp image are $2^8 = 256$, the 8bpp images were always generated by their 16bpp counterparts by normalizing the intensity values:

$$I_{8bpp} = 255 * I_{16bpp}/65535$$

In the SAMPLING dataset the Grayscale images were obtained by their RGB counterparts by applying the SMPTE 295M-1997 standard [SMP97] coefficients to the RGB intensities:

$$Gray = (0.2126 * Red) + (0.7152 * Green) + (0.0722 * Blue)$$

The images were saved using the patent-free Portable Network Graphics (PNG) format [Roe, ISO04, AWTT]. This standard format offers lossless compression and support for a wide range of image types. Since the PNG format supports different compression methods, the free software application OptiPNG (Advanced PNG Optimizer) [Opt] was used to optimize and compress the image files to a smaller size, without losing any information. The OptiPNG option "-nx" were always used to avoid performing any reduction (bit-depth, color, palette). The optimization parameters chosen for the various categories were: "-o5" for SAMPLING, "-zc9 -zm8 -zs1 -f5" for COLOR and "-zc9 -zm8 -zs0 -f0" for PATTERNS and SAMPLING_PATTERNS. These settings were chosen mainly as a good compromise between compression ratio and speed, since compressing more than 2 million images may take a considerable amount of time.

In the PATTERNS dataset, to achieve a good quality anti-aliasing effect, the intensity value of the pixels is always proportional to the portion of the pixel covered by the maximum intensity surface.

3. Archive organization

The TESTIMAGES archive is publicly available at the web address http://testimages.tecnick.com. The archive is currently organized in four main sections: SAMPLING and SAMPLING_PATTERNS (mainly aimed at testing resampling algorithms), COLOR (mainly aimed at testing color rendering on different displays) and PATTERNS (mainly aimed at testing the rendering of standard geometrical patterns). Despite their main categorization, the images in the archive can be easily repurposed for different scenarios. The images are generally organized in folders and subfolders, where every folder indicates one common feature of the contained images (i.e. bit-depth, number of channels, resolution). Due to the large size of the archive, the images are aggregated for common features (".tar" file extension) [GNUb] and compressed using the bzip2 algorithm (".bz2" file extension) [bzi].

3.1. Naming conventions

The file names of the individual images are designed to follow a common pattern:
img_[W]x[H]_[NCH]x[BPP]bit_[EX].png
where:

- **[W]** : image width in pixels or number of image columns
- **[H]** : image height in pixels or number of image rows
- **[NCH]** : number of channels or primary colors (i.e.: 1 for grayscale, 3 for RGB)
- **[BPP]** : bit-per-pixel for each channel (also know as color-depth or bit-depth)
- **[EX]** : extra information used to identify the specific image, as described in the later chapters

This common pattern makes it easy to select images with specific charateristics. In the following chapters we will only refer to the **[EX]** portion of the file name.
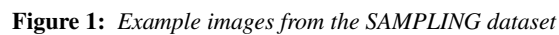
4. SAMPLING dataset

The main purpose of this dataset is providing natural images to test resampling algorithms (i.e. interpolation, zooming, enlargment and superresolution). This dataset uniquely provides a base of 40 RBG 1200x1200 pixels reference images with a bit-depth of 16bpp and high dynamic range (HDR). The reference images were processed to obtain different sub-resolutions and variation with 8bpp and grayscale, for a total of more 220K images.

The test images were chosen to be square for simplicity and symmetry. The value 1200 was chosen because it is approximately half of the vertical resolution of a common 10 Megapixel SLR Camera, it is the vertical resolution of standard UXGA displays and can be divided by 2, 3, 4, 5, 6, 8, 10 and 12 (reduction factors).

4.1. Generating 16bpp reference images

Since currently there are no 16bpp capable photographic cameras in the consumer market, to acquire 16bpp natural images an approach involving multiple exposures was used. The downside of the multi-exposure approach is that it limits the subjects that can be photographed to static objects.

The subjects were photographed using a *Canon 400D* Compact SLR camera mounted on a tripod. This camera is able to produce images in RAW format with a resolution of 3888x2592 pixels and 12bpp. To overcome the standard camera limitation of 3 shots when using the *Automatic Exposure Bracketing* mode (AEB), a modified version of the 400plus firmware was compiled and installed [Asu, 400]. The camera was set to shoot in RAW format using the *Extended AEB* (EAEB) mode and the Aperture Priority to keep the optical parameters invariant between the exposures. The lens was set to minimal aperture (F32) to increase the *depth-of-field* (DOF). The EAEB mode was set with a sequence

Figure 1: *Example images from the SAMPLING dataset*

of 25 different exposures with a separation of 1/3EV to get a total Exposure Value variation of 8EV (from -4EV to +4EV).

Once all the various exposures were acquired, the RAW images were converted to 16bit TIFF format using the camera software and aligned (registered) using the **align_image_stack** command-line tool available in the development version of hugin [Pan]. While not strictly necessary, the alignment step was introduced to compensate for the small variations caused by the camera mirror movement.

A custom Octave/MATLAB software script was then used to merge the different exposures and generate the reference images by: cropping the central part of each exposure to 2448x2448 pixels; summing the contribution of every intensity level (the simple summation combines all the information in the individual exposures without loss [GN03]); reducing the image size to 1224x1224 pixels using the algorithm described below; normalizing the intensity values of the resulted image to a maximum of $(2^{24} - 1)$; applying the Automatic White Balance (AWB) algorithm proposed by Wang et al. [WZDZ11], with a threshold of 1%; normalizing and saving the image to 16bpp (maximum intensity level $= 2^{16} - 1 = 65535$).

At this point 40 16bpp RGB "uncropped" images with a resolution of 1224x1224 pixels were available. A border of 12 pixels was further cropped from these images to get the reference resolution of 1200x1200 pixels, so the top left pixel in the reference image correspond to the pixel in position (13,13) of the uncropped image.

4.2. Resized versions

For each reference image, additional resized images were generated.

The intensity value of each pixel in the resized image was obtained by averaging the values of RxR sub-arrays (where R is the reducing factor).

The reduction factors were chosen to perfectly divide the 1200 width and height values of the reference images (Table 1).

The resized images were always generated starting from the uncropped image but always cosidering a sub-sampling mask of 1200x1200 pixels at the time. For each resizing resolution, all the possible variations obtained by moving (shifting) the sub-sampling mask were saved. The number of possible shift variations for a given reduction factor R are: $((2 \times R) - 1)^2$. The image names contain a shifting indicator, a string composed by two groups of one letter and two

Reduction factor	Resolution [Width x Height]	Variations (shifts)
2	600x600	9
3	400x400	25
4	300x300	49
5	240x240	81
6	200x200	121
8	150x150	225
10	120x120	361
12	100x100	529

Table 1: *Reduction factors of the SAMPLING images*

numbers. The letters indicates the direction of the shift (C: center, T: top, B: bottom, L: left, R: right) while the numbers indicate the number of pixels. The images aligned with the reference image, or the pixel in position (13,13) of the uncropped image, are identified by the "C00C00" shifting indicator.

The shifting variations can be used to compensate the pixel shift introduced by some zooming algorithms. This is a common problem, which arises, for example, when comparing hole-filling interpolation methods with uniform-area resampling methods and more often than not is handled incorrectly. They can also be used to test classical SuperResolution (SR) algorithms, reconstructing the high-resolution reference image from the shifted low-resolution ones.

5. SAMPLING_PATTERNS dataset

This is an extension of the SAMPLING archive containing 424 artificial Grayscale reference images, generated as described in the following PATTERNS section for the 1224x1224 pixels resolution, for a total of more than 1 million image variations. The main purpose of this dataset is to test in isolation specific features of resampling algorithms.

6. COLOR dataset

The main purpose of this dataset is providing images to test the color rendering on different displays and facilitate color adjustments.

This and the PATTERNS archive contain 8bpp and 16 bpp (bit-per-pixel) 3-Channel Red-Green-Blue (RGB) images for the following 114 standard and non-standard resolutions for computer monitors, televisions, cinema projectors, mobile phone screens, camera sensors and other generic displays:
100x100, 120x120, 150x150, 160x120 QQVGA, 160x128, 160x144, 200x160, 200x200, 208x176, 220x176, 226x170, 230x172, 240x160 HQVGA, 240x180, 240x240, 300x300, 320x200 CGA, 320x240 QVGA, 320x256, 352x288 CIF, 360x240 WQVGA, 384x240 WQVGA, 384x288 SIF,

400x240 WQVGA, 400x400, 416x352, 420x292, 432x240 FWQVGA, 480x272 WQVGA, 480x320 HVGA, 480x360, 512x342, 512x384, 528x436, 600x600, 640x200, 640x256, 640x350, 640x360 nHD, 640x400, 640x480 VGA, 640x512, 720x348 Hercules, 720x350 MDA, 720x480 WVGA, 720x576, 768x576 PAL, 800x480 WVGA, 800x600 SVGA, 832x624 SVGA, 854x480 FWVGA, 864x480, 960x540 qHD, 960x640 DVGA, 984x738, 1024x480, 1024x600 WSVGA, 1024x768 XGA, 1136x640, 1152x768 WXGA, 1152x864 XGA+, 1152x870, 1152x900, 1200x824, 1200x1200, 1280x720 HD720, 1280x768 WXGA768, 1280x800 WXGA800, 1280x854, 1280x960 SXGA-, 1280x1024 SXGA, 1360x768, 1366x720, 1366x768, 1440x900 WXGA+, 1440x960 WSXGA, 1400x1050 SXGA+, 1600x768, 1600x900 HD+, 1600x1200 UXGA, 1680x944, 1680x1050 WSXGA+, 1920x1080 FHDTV, 1920x1200 WUXGA, 1920x1440, 2048x1050, 2048x1080 2K DC, 2048x1152 QWXGA, 2048x1536 QXGA, 2560x1440 WQHD, 2560x1600 WQXGA, 2560x2048 QSXGA, 2880x1620, 2880x1800 QWXGA+, 3200x1800 WQXGA+, 3200x2048 WQSXGA, 3200x2400 QUXGA, 3840x2160 4K SHDTV, 3840x2400 WQUXGA, 4096x2160 4K DC, 4096x2560 4K, 4096x3072 HXGA, 5120x2880 UHD+, 5120x3200 WHXGA, 5120x4096 HSXGA, 6400x4096 WHSXGA, 6400x4800 HUXGA, 7680x4320 8K FUHDTV, 7680x4800 WHUXGA, 8192x4320 8K DC, 8192x4608, 10000x7000 IMAX, 15360x8640 16K QUHDTV, 16384x8640 16K DC.

These resolutions include the ones used for the SAMPLING dataset and the ones proposed as standards for very high resolution television and digital cinema. Examples images of this dataset are shown in Figure 2.

For every resolution listed above and for both 8bpp and 16bpp, the following 7 images were generated:

- **color_bars_CMYKWRGB** : Eight vertical bars at maximum intensity (255 for 8bpp and 65535 for 16bpp) representing different colors in the following order: Cyan (Green + Blue), Magenta (Red + Blue), Yellow (Red + Green), Black, White (Red + Green + Blue), Red, Green, Blue.
- **color_bars_CMYKWRGB_100IRE** : Variation of the color_bars_CMYKWRGB where the minimum intensity level is set to 7.5IRE (16 for 8 bit, 4096 for 16 bit) and the maximum intensity level is set to 100IRE (235 for 8 bit, 60160 for 16 bit). For explanation of IRE see SMPTE 170M:2004 [SMP04].
- **color_bars_CMYKWRGB_75IRE** : Variation of the color_bars_CMYKWRGB where the minimum intensity level is set to 7.5IRE (16 for 8 bit, 4096 for 16 bit) and the maximum intensity level is set to 75IRE (180 for 8 bit, 46144 for 16 bit). For explanation of IRE see SMPTE 170M:2004 [SMP04].
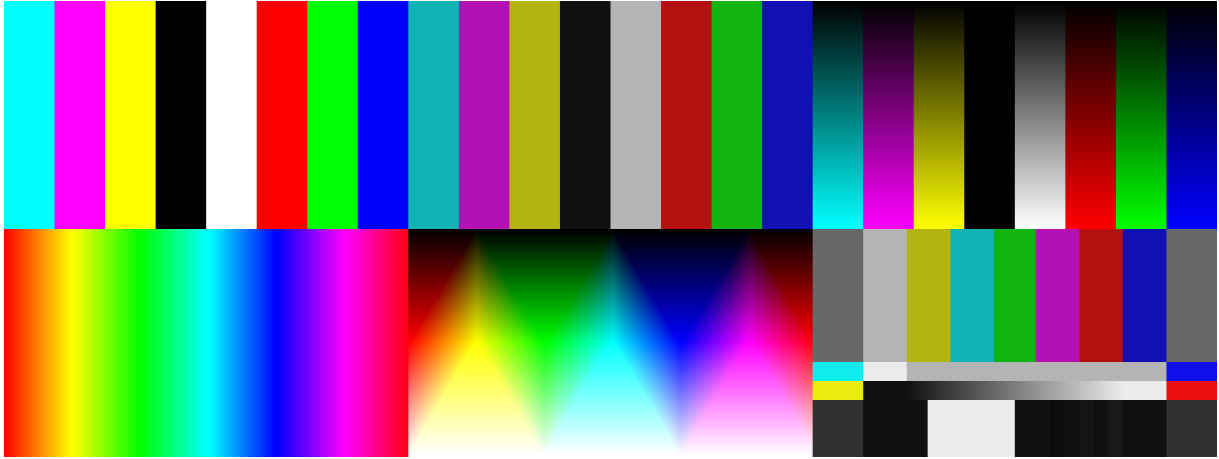- **color_bars_CMYKWRGB_gradient** : Variation of

Figure 2: *Example images from the COLOR dataset*

the color_bars_CMYKWRGB where intensity levels varies linearly from top 0% to bottom 100%.

- **color_rainbow** : Six vertical color sections with the following intensity variations to obtain a "rainbow" effect: 1. *red = MAX*, *green = 0 → MAX*, *blue = 0*; 2. *red = MAX → 0*, *green = MAX*, *blue = 0*; 3. *red = 0*, *green = MAX*, *blue = 0 → MAX*; 4. *red = 0*, *green = MAX → 0*, *blue = MAX*; 5. *red = 0 → MAX*, *green = 0*, *blue = MAX*; 6. *red = MAX*, *green = 0*, *blue = MAX → 0*.

- **color_rainbow_gradient** : Variation of the color_rainbow where intensity levels varies linearly to obtain a black to white variation from top to bottom.

- **color_SMPTE_RP_219_2002** : This image is generated as described in SMPTE RP 219:2002 [SMP02] and resized to perfectly match different resolution. The Color bar signal is generated with unconventionally slow rise and fall time value to facilitate video level control and monitor color adjustments.

7. PATTERNS dataset

The main purpose of this dataset is providing images to test the rendering of standard geometrical patterns by displays, monitors, televisions and projectors. These patterns facilitate the adjustment of geometrical parameters and helps identify defects of the rendering device. The broad range of simple geometrical patterns also helps to optimize or identify specific issues in some image processing algorithms. For example, by verifying if the patterns are preserved when applying an image compression algorithm, by testing the effects of different texture filtering and resampling approaches programmed in shaders, or by evaluating the printing quality of a particular device.

This archive contains 8bpp and 16bpp (bit-per-pixel) 3-Channel Red-Green-Blue (RGB) images for the 114 standard and non-standard resolutions previously listed in the

COLOR dataset. Examples images of this dataset are shown in Figure 3.

For every resolution listed above, the following image variations were generated 4:

- **GRAY**: (1-Channel Grayscale) These are the reference images containing the geometric patterns as described below. These images are then used to generate the following 3-Channel RGB variations by copying the grayscale layer in one or more color channel: **RED**, **GREEN**, **BLUE**, **RED_GREEN**, **RED_BLUE**, **GREEN_BLUE**, **RED_GREEN_BLUE**.

For every resolution, color variation and for both 8bpp and 16bpp, the following image types and variations were generated:

- **bars_[ORIENT]_[TYPE]_[BW]_[PHASE]xhalfPI** : Constant size bars with the following characteristics: **[ORIENT]** bars orientation (vertical, horizontal); **[TYPE]** intensity curve perpendicular to the bar orientation (square = step curve; sin = sinusoidal; triangle = sawtooth wave); **[BW]** witdh of an entire curve period - the bars period is always a power of 2; **[PHASE]** the curve phase expressed in number of $\pi/2$ (0-3).

- **bars_[ORIENT]_[TYPE]_progressive** : Variable size bars. The smallest bars starts at the image center and increase their size as power of 2. The variation option are: **[ORIENT]** bars orientation (vertical,horizontal); **[TYPE]** intensity curve perpendicular to the bar orientation (square = step curve; sin = sinusoidal; triangle = sawtooth wave).

- **grid_square_progressive** : Grid of rectangles obtained by combining (XOR) the bars_horizontal_square_progressive and bars_vertical_square_progressive images.
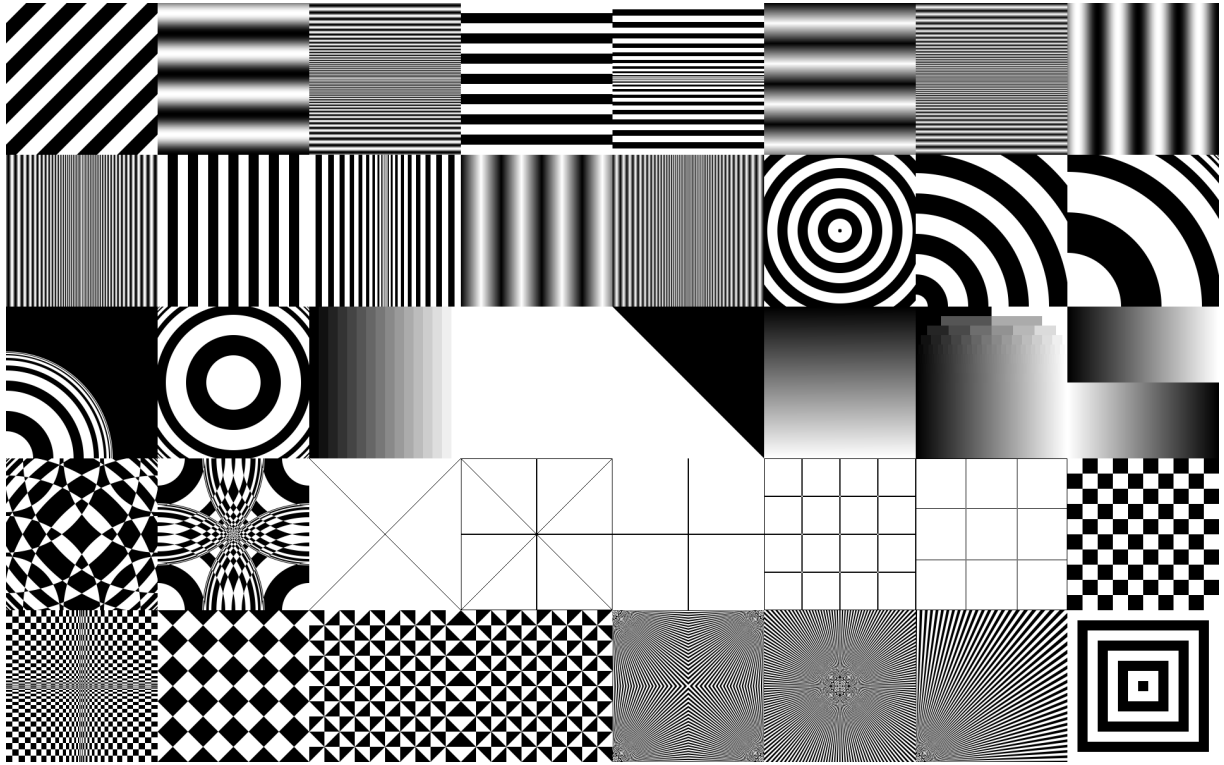
- **bars_45deg_tlbr_[BW]** : Solid bars oriented at 45

Figure 3: *Example images from the PATTERNS dataset*

degrees, from top-left to bottom-right. **[BW]** is the witdh of an entire curve period; the bars period is always a power of 2.

- **bars_45deg_bltr_[BW]** : Solid bars oriented at 45 degrees, from bottom-left to top-right. **[BW]** is the witdh of an entire curve period; the bars period is always a power of 2.
- **rectangles_concentric_[BW]_0** : Concentric rectangles. **[BW]** is the witdh of the rectangle border (power of 2).
- **rectangles_concentric_[BW]_1** : Inverse of rectangles_concentric_[BW]_0.
- **grid_rectangle_[BW]x[BH]_0** : Grid with alternate black and white rectangles with the **[BW]** width and **[BH]** height. The rectangles dimensions **[BW]** and **[BH]** are chosen to perfectly fit the available image space. The image width is a multiple of 2 x **[BW]** and the image height is a multiple of 2 x **[BH]**.
- **grid_rectangle_[BW]x[BH]_1** : Inverse of grid_rectangle_[BW]x[BH]_0.
- **diagonal_[CORNER]** : Diagonal that divides the images in black and white parts. The **[CORNER]** values are: tl=top-left, tr=top-right, br=bottom-right, bl=bottom-left.
- **grid_triangle_a_[BW]x[BH]_0** : Grid obtained by combining triangles to form black and white rhombus.

The rectangles dimensions **[BW]** and **[BH]** are chosen to perfectly fit the available image space, the image width is a multiple of 2 x **[BW]** and the image height is a multiple of 2 x **[BH]**.

- **grid_triangle_a_[BW]x[BH]_1** : Inverse of grid_triangle_a_[BW]x[BH]_0.
- **grid_triangles_b_[BW]x[BH]_0** : Grid obtained by combining triangles in alternate black and white positions. The rectangles dimensions **[BW]** and **[BH]** are chosen to perfectly fit the available image space. The image width is a multiple of 2 x **[BW]** and the image height is a multiple of 2 x **[BH]**.
- **grid_triangles_b_[BW]x[BH]_1** : Inverse of grid_triangles_b_[BW]x[BH]_0.
- **grid_lines_cross_0** : Central white cross in a black background. The line width is always 2 pixels wide to result perfectly centered when the correspondent resolution dimension is even.
- **grid_lines_cross_1** : Inverse of grid_lines_cross_0.
- **grid_line_45deg_0** : Grid obtained by combining (XOR) the four 45 degrees diagonal surfaces crossing each image corner.
- **grid_line_45deg_1** : Inverse of grid_line_45deg_0.
- **grid_lines_quarters_0** : Grid obtained by divid-

ing the image in 4x4 rectangles using white lines over a black background.

- **`grid_lines_quarters_1`** : Inverse of grid_lines_quarters_0.
- **`grid_line_thirds_0`** : Grid obtained by dividing the image in 3x3 rectangles using white lines over a black background.
- **`grid_line_thirds_1`** : Inverse of grid_line_thirds_0.
- **`grid_line_intersections_0`** : Grid obtained using four 45 degrees lines passing in each image corner, and horizontal and vertical lines passing in each intersection.
- **`grid_line_intersections_1`** : Inverse of grid_line_intersections_0.
- **`rays_radial_corner_[CORNER]`** : Alternating black and white ray lines starting at image edge positions covering an angle of 1 degree each. The **`[CORNER]`** values are: tl=top-left, tr=top-right, br=bottom-right, bl=bottom-left, t=top, b=bottom, l=left, r=right.
- **`rays_radial_allcorners`** : Grid obtained combining (XOR) the images rays_radial_corner_tl, rays_radial_corner_tr, rays_radial_corner_br, rays_radial_corner_bl.
- **`rays_radial_center`** : Alternating black and white ray lines starting at image center covering an angle of 1 degree each.
- **`circles_corner_[CORNER]_[BW]`** : Alternating black and white concentric circles with constant **`[BW]`** tickness, centered in one of the image corners. The **`[CORNER]`** values are: tl=top-left, tr=top-right, br=bottom-right, bl=bottom-left, t=top, b=bottom, l=left, r=right.
- **`circles_progressive_a_corner_[CORNER]`** : Alternating black and white concentric circles with variable tickness, centered in one of the image corners.
- **`circles_progressive_b_corner_[CORNER]`** : Variation of the circles_progressive_a_corner_[CORNER].
- **`grid_circles_progressive_a`** : Grid obtained combining (XOR) all the circles_progressive_a_corner_[CORNER] images.
- **`grid_circles_progressive_b`** : Grid obtained combining (XOR) all the circles_progressive_b_corner_[CORNER] images.
- **`circles_center_[BW]`** : Alternating black and white concentric rings with constant (**`[BW]`** / 2) tickness.
- **`gray_level_crossing`** : Uniform intensity transition form zero to maximum value, from left to right in the upper half image and from right to left in the lower half image.
- **`gray_level_bits`** : The vertical space is divided in 16 rows representing intensity quantization from 1 bit (top row - 2 values) to 16 bit (bottom row - 65536 values). In the 8bpp variation of this image, due to the normalization, the rows from 9 to 16 are identical to the row 8.

- **`gray_level_all`** : Uniform transition from zero to maximum intensity, distribuited across all pixels from left to right and from top to bottom.
- **`color_full`** : All pixels are set to the maximum intensity value.
- **`color_16th`** : The horizontal space is divided in 16 vertical bars representing intensity variation from zero (black) to maximum value (white).

Figure 4: *Example of image color variations of a single geometrical pattern.*

8. Differences with other archives and licensing

Large datasets of images are currently available on the Internet, most of which are not easily usable for testing due to licensing and copyright issues. Unfortunately, the ones that contain free-license images (e.g. http://commons.wikimedia.org/) often do not include specific images for testing displays and image processing algorithms. Other image datasets, commonly used for testing basic image processing tools, are usually old and contain a limited variety of resolutions, color bands, gradients and regular patterns. Classical examples of these datasets are http://sipi.usc.edu/database and http://www.imageprocessingplace.com/. Other very good archives are available for testing specific high-level task algorithms (e.g. stereo matching, optical flow, object recognition, texture analysis, etc.), however they are not generally suitable for basic tasks.

TESTIMAGES is therefore the first large dataset for the research tasks proposed here, and presents original features that are not found in other public archives of images. For example, the availability of images with a large number of different resolutions, the availability of original 16 bit per color depth images or the availability of shifted downsampled images.

Another relevant feature of this archive is related to its license of use. To avoid common issues related to the use of images with unknown origin or protected by restrictive copyright or license, all the images in the TESTIMAGES archive are copyrighted by the author and distributed under

the Creative Commons license *Attribution-NonCommercial-ShareAlike 4.0 International* (CC BY-NC-SA 4.0), whose full terms are available at `http://goo.gl/8TfzaH`.

In summary the license states that you are free to *Share* (copy and redistribute the material in any medium or format) and *Adapt* (remix, transform, and build upon the material) as long as you follow the license terms:

- **Attribution** - You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.
- **NonCommercial** - You may not use the material for commercial purposes.
- **ShareAlike** - If you remix, transform, or build upon the material, you must distribute your contributions under the same license as the original.
- **No additional restrictions** - You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

9. Discussion

Research and development in Computer Graphics and image processing may obtain huge benefit from the availability of public tools for the technical and scientific community. Useful tools go beyond computer code to include test data. The TESTIMAGES archive is proposed as a ready and free to use large image dataset. It includes rich natural images with unusual 16 bit per color depth available at several resolution and downscaling options, color and geometrical patterns also available in a large number of variations and resolutions. Each archive section has been designed for specific purposes. For example, the SAMPLING dataset has been primarily designed as a tool to help people develop zooming and downlscaling methods, the PATTERNS dataset can be extremely useful for evaluating texture filtering and compression methods, etc. However, the peculiar image features makes the data valuable for evaluating different kinds of image processing algorithms not considered by the author. As an example of this, some of the first images used to populate the archive, even if no information on this was given to the scientific community, have already been used as test images for completely different algorithm types as watermarking [ZPP12] or denoising in compressive sensing [SGPA12].

References

[400] 400PLUS: 400plus - a firmware hack for the canon 400d / xti digital camera. Retrieved June 10, 2014, from `http://goo.gl/GL8FF6`. 2

[AG08] ASUNI N., GIACHETTI A.: Accuracy improvements and artifacts removal in edge based image interpolation. In *VISAPP (1)* (2008), pp. 58–65. 1

[Asu] ASUNI N.: 400plus extending the eaeb (extended automatic exposure bracketing). Retrieved June 10, 2014, from `http://goo.gl/x1iXQ1`. 2

[AWTT] ARL-WMRD-TED-TIB G. R.-P.: Registration of new media type image/png. Retrieved June 10, 2014, from `http://goo.gl/E8ykL6`. 2

[bzi] BZIP.ORG: bzip2. Retrieved June 10, 2014, from `http://www.bzip.org`. 2

[GN03] GROSSBERG M. D., NAYAR S. K.: High dynamic range from multiple images: Which exposures to combine. In *Proc. ICCV Workshop on Color and Photometric Methods in Computer Vision (CPMCV)* (2003). 3

[GNUa] GNU: Gnu octave. Retrieved June 10, 2014, from `http://goo.gl/tXQVds`. 1

[GNUb] GNU: Gnu tar. Retrieved June 10, 2014, from `http://goo.gl/5tNmhc`. 2

[ISO04] ISO/IEC: Iso/iec 15948:2004 - information technology – computer graphics and image processing – portable network graphics (png): Functional specification, 2004. URL: `http://goo.gl/GkRavd`. 2

[KSP13] KOPF J., SHAMIR A., PEERS P.: Content-adaptive image downscaling. *ACM Transactions on Graphics (TOG) 32*, 6 (2013), 173. 1

[MWDG13] MASIA B., WETZSTEIN G., DIDYK P., GUTIER-REZ D.: A survey on computational displays: Pushing the boundaries of optics, computation, and perception. *Computers & Graphics 37*, 8 (2013), 1012–1038. 1

[NM14] NASROLLAHI K., MOESLUND T. B.: Super-resolution: A comprehensive survey. *Machine Vision & Applications* (2014). 1

[Opt] OPTIPNG: Optipng: Advanced png optimizer. Retrieved June 10, 2014, from `http://goo.gl/kP91`. 2

[Pan] PANOTOOLS: Panotools - align image stack. Retrieved June 10, 2014, from `http://goo.gl/EhZFI2`. 3

[Roe] ROELOFS G.: Png - portable network graphics - an open, extensible image format with lossless compression. Retrieved June 10, 2014, from `http://goo.gl/RWjFD`. 2

[SGPA12] SERMWUTHISARN P., GANSAWAT D., PATANAVI-JIT V., AUETHAVEKIAT S.: Impulsive noise rejection method for compressed measurement signal in compressed sensing. *EURASIP Journal on Advances in Signal Processing 2012*, 1 (2012), 1–23. 8

[SMP97] SMPTE: Smpte st 295:1997 - for television - 1920 ÃŮ 1080 50 hz - scanning and interfaces, 1997. URL: `http://goo.gl/IvqyxT`, `doi:10.5594/S9781614824336`. 2

[SMP02] SMPTE: Smpte rp 219:2002 - high-definition, standard-definition compatible color bar signal, 2002. URL: `http://goo.gl/RrmnZP`, `doi:10.5594/S9781614822301`. 5

[SMP04] SMPTE: Ansi/smpte 170m-2004 for television âĂŤ composite analog video signal âĂŤ ntsc for studio applications, 2004. URL: `http://goo.gl/s4wkny`, `doi:10.5594/S9781614823353`. 4

[WZDZ11] WANG S., ZHANG Y., DENG P., ZHOU F.: Fast automatic white balancing method by color histogram stretching. In *Image and Signal Processing (CISP), 2011 4th International Congress on* (2011), vol. 2, IEEE, pp. 979–983. 3

[ZPP12] ZIGOMITROS A., PAPAGEORGIOU A., PATSAKIS C.: Social network content management through watermarking. In *Trust, Security and Privacy in Computing and Communications (TrustCom), 2012 IEEE 11th International Conference on* (2012), IEEE, pp. 1381–1386. 8