**NIST Technical Note 2207**

# Gauging the Difficulty of Image Segmentation

Marek Franaszek

This publication is available free of charge from:
https://doi.org/10.6028/NIST.TN.2207

National Institute of
Standards and Technology
U.S. Department of Commerce

**NIST Technical Note 2207**

# Gauging the Difficulty of Image Segmentation

Marek Franaszek
*Intelligent Systems Division*
*Engineering Laboratory*

This publication is available free of charge from:
https://doi.org/10.6028/NIST.TN.2207

March 2022


U.S. Department of Commerce
*Gina M. Raimondo, Secretary*

National Institute of Standards and Technology
*James K. Olthoff, Performing the Non-Exclusive Functions and Duties of the Under Secretary of Commerce for Standards and Technology & Director, National Institute of Standards and Technology*

**National Institute of Standards and Technology Technical Note 2207**
**Natl. Inst. Stand. Technol. Tech. Note 2207, 39 pages (March 2022)**
**CODEN: NTNOEF**

**This publication is available free of charge from:**
**https://doi.org/10.6028/NIST.TN.2207**

**Abstract**

Image segmentation is the first step in a complex process of object recognition. This report presents a method to gauge the difficulty of segmentation by calculating a scalar parameter *Q* for an image. This parameter depends on a distribution of the intensity of the grayscale image and the distribution of the clustering of pixels. It is assumed that images with a smaller number of clusters are easier to segment than images with a larger number of clusters. Since segmentation precedes any human perception and categorization, the distribution of parameter *Q* introduced in this study may be useful in characterizing the variability of images collected in a training dataset for the development of object recognition algorithms which use machine learning (ML) methods. Parameter *Q* can be especially useful for building a representative dataset of images for training ML algorithms. To demonstrate a link between particular values of *Q* and different segmentation conditions, a few grayscale images were distorted by some common transformations (such as Gaussian noise, median filtering, and decrease of color depth) and the corresponding values of parameter *Q* were calculated. To demonstrate a possible use of the parameter *Q* on data other than grayscale images, depth images of flat planar targets taken by two depth cameras were also processed.

**Key words**

Dataset of 2D images, image segmentation, object recognition, training machine learning algorithms.

**Table of Contents**



**List of Tables**

**List of Figures**

## 1. Introduction

The development of predictive models for object recognition from 2D images requires large datasets of images on which machine learning (ML) algorithms can be trained. Usually, categories of objects are initially predetermined in datasets used to train ML algorithms, as it helps to develop specialized models which perform better within a specific domain. One example is the Common Objects in Context (COCO) dataset, which contains 118,287 training images covering 80 different categories [1]. Building such a dataset from a universe of 2D images requires carefully-predetermined selection criteria so that the resulting dataset is well-balanced (i.e., all preselected categories of interest are well represented in the dataset). The old question of which is more important, more data or better algorithms, remains open for debate [2], but collecting as many images as possible seems to be a commonly-accepted requirement for achieving a well-performing algorithm on test cases.

Despite the previous trend of using carefully-curated datasets, recently there has been a growing interest in deliberately using unbalanced training datasets, in which the images in each category are not evenly distributed. This trend reflects an observation that a large dataset with a limited number of carefully chosen categories also contains many other categories which are not used in training. To increase the number of categories used in the training dataset and to build a model which can recognize objects in a larger domain of categories, new images need to be added to ensure a well-balanced dataset. However, the addition of new images to the selected categories introduces new underrepresented categories. Thus, the problem of having a long tail distribution of categories and unbalanced datasets persists as the Large Vocabulary Instance Segmentation (LVIS) dataset shows [3]. The LVIS is based on the original COCO dataset, with the addition of almost 1,000 new categories resulting from annotating everything that can be segmented in the original COCO dataset.

This process of gauging the quality of a training dataset is based on human perception and categories determined by humans (it's no surprise that categories in the LVIS were derived from synsets of the WordNet dictionary [4]). An alternative approach would involve assigning to each image a scalar parameter which is based solely on image processing techniques, without involving human perception. Designing such objective parameters is the domain of Image Quality Assessment (IQA) [5]. The techniques developed in this domain are used to evaluate image compression algorithms or the deterioration of image quality in transmitting channels. As such, they are suited for performing a relative comparison of a distorted image with the original image [6] and not for characterizing the variability of images in a large dataset. Even though a great effort has been made to make IQA objective, the ultimate evaluation of these procedures is gauged against a cohort of human observations (impressions), and different procedures may yield results conflicting with the opinions of experienced readers [7]. Another example of a parameter that can characterize a single image is the image entropy $E$ based on the well-known Shannon formula

$$E = -\sum_i p(i) \log_2 p(i) \ , \tag{1}$$

where $p(i)$ is the fraction of pixels belonging to the *i*-th category and the summation is over all categories. To avoid the subjectivity of human defined categories, the summation in Eq.(1) could be over all intensities $i = I_{min}, \dots, I_{max}$ represented in the grayscale image. While such entropy provides a useful measure of disorder, it is not sensitive to the clustering of pixels with similar intensities in the image. The process of object recognition starts with finding these clusters and then analyzing them. This first step is common to all recognition algorithms and is achieved by image segmentation [8, 9]. There are objective entropy-based measures for gauging image segmentation, but similarly to IQA methods, these measures are also designed for the relative comparison of two segmentation procedures (or two outcomes of the same procedure executed with two different input parameters) [10]. Therefore, a normalized scalar parameter $0 \leq Q \leq 1$ (derived from unscaled parameter $q$ introduced in this report) is designed to be sensitive to a degree of pixel clustering and can be interpreted as a metric that gauges the difficulty of the segmentation process. This metric is based on the assumption that an image containing fewer clusters is easier to segment than an image with many clusters scattered throughout the image area. The rationale for this assumption comes from the fact that incorrect segmentation of a particular cluster of pixels (i.e., either its over- or under-segmentation) is more likely to happen if there are other relevant clusters very close to it. With a fixed image size, such configurations are more likely when a larger number of clusters coexist and thus, the segmentation process is characterized as being more difficult. The distribution of $q$ for a large scale dataset of images could provide useful insight into the variability of images included in the dataset, thereby avoiding the otherwise unavoidable trap of an unbalanced, long tail distribution of human-perceived categories. This report presents a definition and calculation of $q$ for a few example images. Each original image was modified by a few commonly-used transformations (e.g., adding noise, median filtering, and decrease of color depth) to see their impact on the calculated $q$.

## 2. Characterizing 2D images using parameter $q$

Let $I(x, y)$ be an image of size $(N_x, N_y)$ where $x = 1, \dots, N_x$ , $y = 1, \dots, N_y$ and its $N_I$ discrete scalar values are $I_{min} \leq I \leq I_{max}$. The most common example is a grayscale image acquired with a digital camera: the size of the image is determined by a resolution of the camera sensor (charge-couple device CCD or complementary metal oxide semiconductor CMOS) and a range of intensity values by the camera analog-to digital (A/D) converter (commonly used 8 bits converters yield $N_I = 256$ and $[I_{min}, I_{max}] = [0, 255]$). Another example would be a depth image where each pixel stores a distance from the camera's focal plane to a point in the acquired physical scene (or 0 if no distance can be obtained for particular pixels). Generally, the parameter $q$ introduced in this report can be calculated for any rectangular table of scalar values $S$. For images that have multiple scalar values at each pixel, e.g., red, green, and blue (RGB) images, a conversion from vector to scalar can be performed before $q$ is calculated, for example color to grayscale transformation [11]

$$I = 0.2989\ R + 0.5870\ G + 0.1140\ B\ , \tag{2}$$

where *R*, *G* and *B* are the intensities in the three color channels at each pixel in the image.

For such 2D scalar images, the parameter $q$ is defined as

$$q = \sum_I N(I)\, D(I)\,, \tag{3}$$

where $D(I)$ is the cumulative fraction of pixels with intensity less than or equal to $I$, and $N(I)$ is the normalized number of clusters resulting from image segmentation with threshold $I$. The summation is over all intensities $I = I_{min}, \ldots, I_{max}$. The cumulative fraction $D(I)$ (which is closely related to the intensity probability distribution $p(I)$) does not require further elaboration as most modern cameras provide the user with the intensity profiles of acquired pictures. However, the number of segmented clusters $N(I)$ needs some explanation, namely how the segmentation is performed and how normalization is done for images with different sizes.

### 2.1. Image segmentation

Thresholding segmentation is a basic procedure which is commonly used as an entry step to more advanced segmentation techniques [12, 13]. Segmentation is based on two concepts: 1) spatial connectivity; 2) acceptance or membership criteria. Two pixels satisfying certain acceptance criteria are connected if they are linked together in the image by pixels that satisfy the acceptance criteria. For two dimensional images, there are two types of connections based on either four or eight nearest neighbors (in this study, the first type is used as shown in Fig. 1). The acceptance condition used in this study requires the intensity of a pixel to be equal or smaller than the threshold intensity $I$. Segmentation starts with finding a pixel which satisfies the condition for $I = I_{min}$. Once such a starting seed is found, the standard region growing procedure begins [13]. For each accepted pixel, its four nearest neighbors are checked, as shown in Fig. 1.

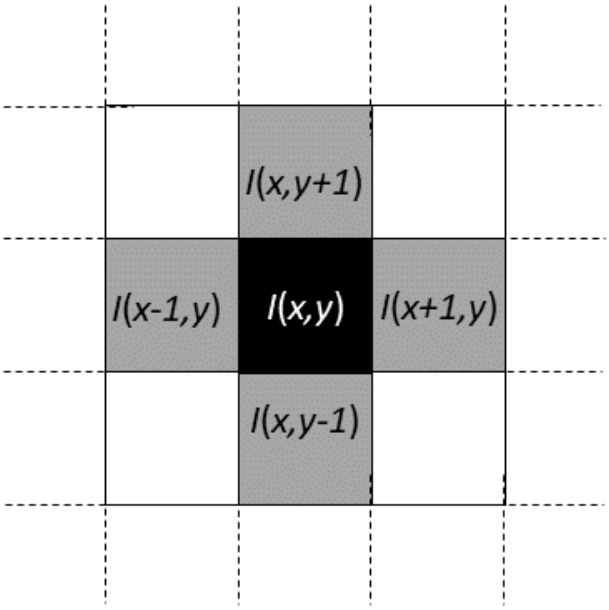


Fig. 1 Illustration of 2D image segmentation: squares represent image pixels, the black square is a starting seed with intensity in the accepted range, and the four gray squares are its nearest neighbors. Gray squares that have intensities in the accepted range become members of the segmented cluster connected to the starting seed, otherwise they are marked as processed non-members. White squares are pixels which have not been processed yet.

If any one of the nearest neighbors has an intensity in the acceptance range, it is added to the list of pixels which are connected to the starting seed, otherwise it is marked as processed. The process continues until exhaustion when no more pixels connected with the starting seed can be accepted. Then, a number of clusters segmented for a current intensity $n(I)$ is increased by one and a search for another unprocessed pixel with intensity equal to or smaller than $I$ continues. Once such a pixel is found, it serves as a new seed for a subsequent region growing procedure. When all pixels in the entire image are processed and no more clusters can be segmented, the whole process repeats for the subsequent values of intensity until $I = I_{max}$. A flowchart illustrating the algorithm is shown in Fig. 2.

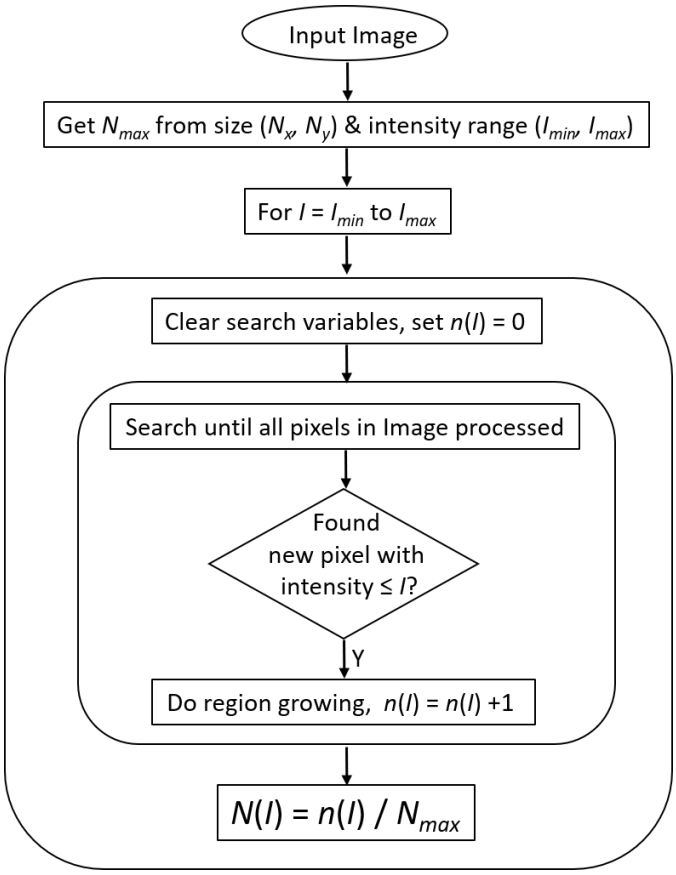


Fig. 2 Flowchart of the algorithm to calculate the normalized number of clusters $N(I)$ for each intensity $I$ represented in the image.

The number of segmented clusters $n(I)$ depends on the cumulative number of all pixels in the whole image $D(I)$ and the spatial distribution of intensity $I(x, y)$ on a 2D plane. To compare the resulting numbers $n(I)$ obtained for images with different sizes $(N_x, N_y)$, the maximum number of disjoint clusters $N_{max}$ should be determined so that the normalized numbers $N(I) = n(I)/N_{max}$ can be calculated. The largest number of clusters can be observed when each cluster consists of only one pixel surrounded by four neighboring pixels which do not satisfy the acceptance criteria. Such a configuration forms the checkerboard pattern shown in Fig. 3. The maximum number of disjoint clusters, $N_{max}$, is determined as

$$N_{max} = \begin{cases} N_x N_y/2 & when\ N_x N_y\ is\ even \\ (N_x N_y + 1)/2 & when\ N_x N_y\ is\ odd \end{cases} \quad (4)$$

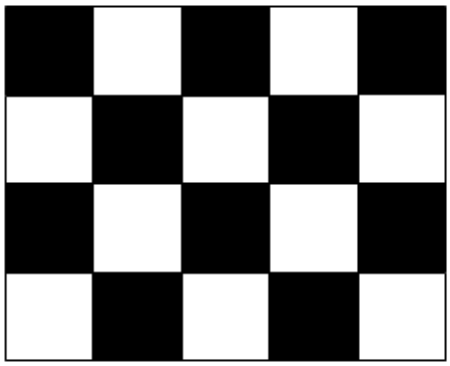

Fig. 3 Checkerboard pattern corresponding to the configuration with the largest possible number of disjoint clusters, $N_{max}$, in a 2D image. In this case, each cluster is formed by only one pixel (the black squares).

The parameter $q$ can then be calculated using Eq.(3). Theoretical bounds for the sum in Eq.(3) are $(0,\ N_I)$, but there are no real 2D images which could yield these extreme values (the minimum would correspond to completely empty image while the maximum could be reached only when all pixels have one intensity). A typical range for $q$ is $(N_I/N_{max}, \sim 10)$.

## 3. Examples of calculating $q$

To get a better understanding of the meaning of different values of $q$, two types of 2D images were processed: 1) an RGB image converted to a grayscale image and 2) a depth image. Each image was transformed using eight different techniques, each of which affected the segmentation process and the corresponding value of $q$. For all eight transformations, $q$ was calculated using the same procedure outlined in the flowchart in Fig.2. All of the used image transformations are briefly described below.

### 3.1. Identity transformation

The original , unchanged grayscale image (converted from RGB using Eq.(2)) or the depth image was used to calculate $q$ using Eq.(3). Segmentation was performed for each grayscale intensity or discrete value of depth present in the image. The raw, unchanged image provided a baseline for parameter $q$.

### 3.2. Grouping locations of pixels with the same intensity

The rectangular 2D array of intensities or depths $I(x,y)$ was reshaped to a 1D vector $I(n)$, where $n = 1, \ldots, N_x N_y$. Then, the elements in the 1D vector were sorted by ascending order, yielding the vector $\tilde{I} = [I_{min}, \ldots, I_{min},\ I_{min}+1, \ldots, I_{min}+1, \ldots,\ I_{max}, \ldots, I_{max}]$ (there are many repeated entries in $\tilde{I}$ since the number of pixels is much larger than intensity depth, $N_x N_y \gg I_{max}$ ). Finally, the vector $\tilde{I}$ was reshaped back to a 2D array $\tilde{I}(x,y)$ of the original size $(N_x, N_y)$ and mapped on a cylinder surface (mapping on a cylinder surface ensures that a segmentation of image $\tilde{I}$ wraps short 1D segments at the boundary of a flat 2D image $I$). The

reshuffled image $\tilde{I}(x,y)$ has exactly the same cumulative fraction of pixels, $D(I)$, and entropy, $E$, calculated in Eq.(1) as the original image $I(x,y)$, but a different number of disjoint clusters: for each intensity, $I$, there is only one cluster, since all intensities with the same value were grouped into one segment in the ordered 1D vector. Therefore, $N(I) = 1/N_{max}$ for all $I_{min} \le I \le I_{max}$. Parameter $q$ calculated in Eq.(3) for the regrouped image is the smallest possible for all images that have the same intensity profile, $D(I)$, and the same size, $(N_x, N_y)$, and is denoted as $q_{min}$.

### 3.3. Salt and pepper noise

Out of $N_I$ intensities, which are spread in the range $(I_{min}, I_{max})$, $N_I/3$ of them are randomly selected. For each selected intensity $I$, all pixels in the image with this intensity had their intensities changed to $I_{max} - I$. The result is that selected bright pixels were flipped to dark and selected dark pixels became bright: hence the name of the transformation which affected both the image profile, $D(I)$, and the distribution of disjoint clusters, $N(I)$.

### 3.4. Random reshuffling of pixel locations

The rectangular 2D array of intensities or depths $I(x,y)$ was reshaped into a 1D vector $I(n)$, where $n = 1, \dots, N_x N_y$. A random perturbation $\tilde{n}$ of integer numbers $[1, \dots, N_x N_y]$ was selected and a new 1D vector of reshuffled intensities $\tilde{I} = I(\tilde{n})$ was created, which was then reshaped back to the 2D array $\tilde{I}(x,y)$ of the original size $(N_x, N_y)$. This transformation preserves the profile of intensity, $D(I)$, of the original image (and, consequently, the entropy $E$ calculated in Eq.(1)), but it changes the number of disjoint clusters, $N(I)$. The resulting distribution, $N(I)$, maximizes the number of disjoint clusters for each intensity. Therefore, the corresponding parameter $q$ has its value close to its largest possible value for the original image. While the exact theoretical upper bound for $q$ is not known, numerical experiments showed that using different random perturbations yielded a series of $q$ values scattered in a very narrow range: the corresponding standard deviation was less than 0.3% of the mean value. Therefore, as an estimate of the largest possible parameter $q$ for all images having the same intensity profile, $D(I)$, and the same size $(N_x, N_y)$, we take

$$q_{max} = (1 + 0.006)\, q\ , \tag{5}$$

where $q$ is calculated for an image transformed by one random perturbation, as described in this section. The estimated upper bound $q_{max}$ together with $q_{min}$ calculated in section 3.2 provide a convenient normalization of parameter $q$ calculated for the original image in section 3.1

$$Q = (q - q_{min})/(q_{max} - q_{min})\ . \tag{6}$$

Normalized $0 \le Q \le 1$ allows direct characterization of images with different sizes, intensity profiles, and semantic contents.

### 3.5. Median filtering

A $5 \times 5$ kernel was used to replace the intensity of each pixel with a median intensity taken from pixels within a kernel centered on a processed pixel. While the image profile, $D(I)$, remained mostly unchanged by this transformation, median filtering had a smoothing effect, which led to a decreased number of segmented disjoint clusters.

### 3.6. Global Gaussian noise

Each pixel in the image was perturbed by zero-mean Gaussian noise with fixed variance $\sigma$. Both $D(I)$ and $N(I)$ were affected by this transformation.

### 3.7. Local Gaussian noise

A subrange $(I_{Lo}, I_{Hi})$ of the full intensity range $(I_{min}, I_{max})$ was selected such that $I_{min} < I_{Lo} < I_{Hi} < I_{max}$. All pixels with their intensities in the selected subrange had their intensities perturbed by zero-mean Gaussian noise with fixed variance $\sigma$. Both $D(I)$ and $N(I)$ were affected by this transformation.

### 3.8. Decrease color depth

Original 8 bytes color depth was reduced to 3 bytes. On grayscale image, intensity $I$ of each pixel was replaced by a reduced intensity $I/2^5$. Both $D(I)$ and $N(I)$ were affected by this transformation.

## 4. Results for grayscale images

Six original RGB images are shown in Fig. 4. Each of them was converted to grayscale using Eq.(2) and processed using the eight methods outlined in sections 3.1 – 3.8. Numerical values of the parameter $q$ are provided in Table 1 and are plotted in Fig. 5. Normalized $Q$ are provided in Table 2 (together with image sizes in pixels) and are plotted in Fig. 6. Processed images and resulting graphs of $D(I)$ and $N(I)$ are displayed in Figs. (7 – 18).

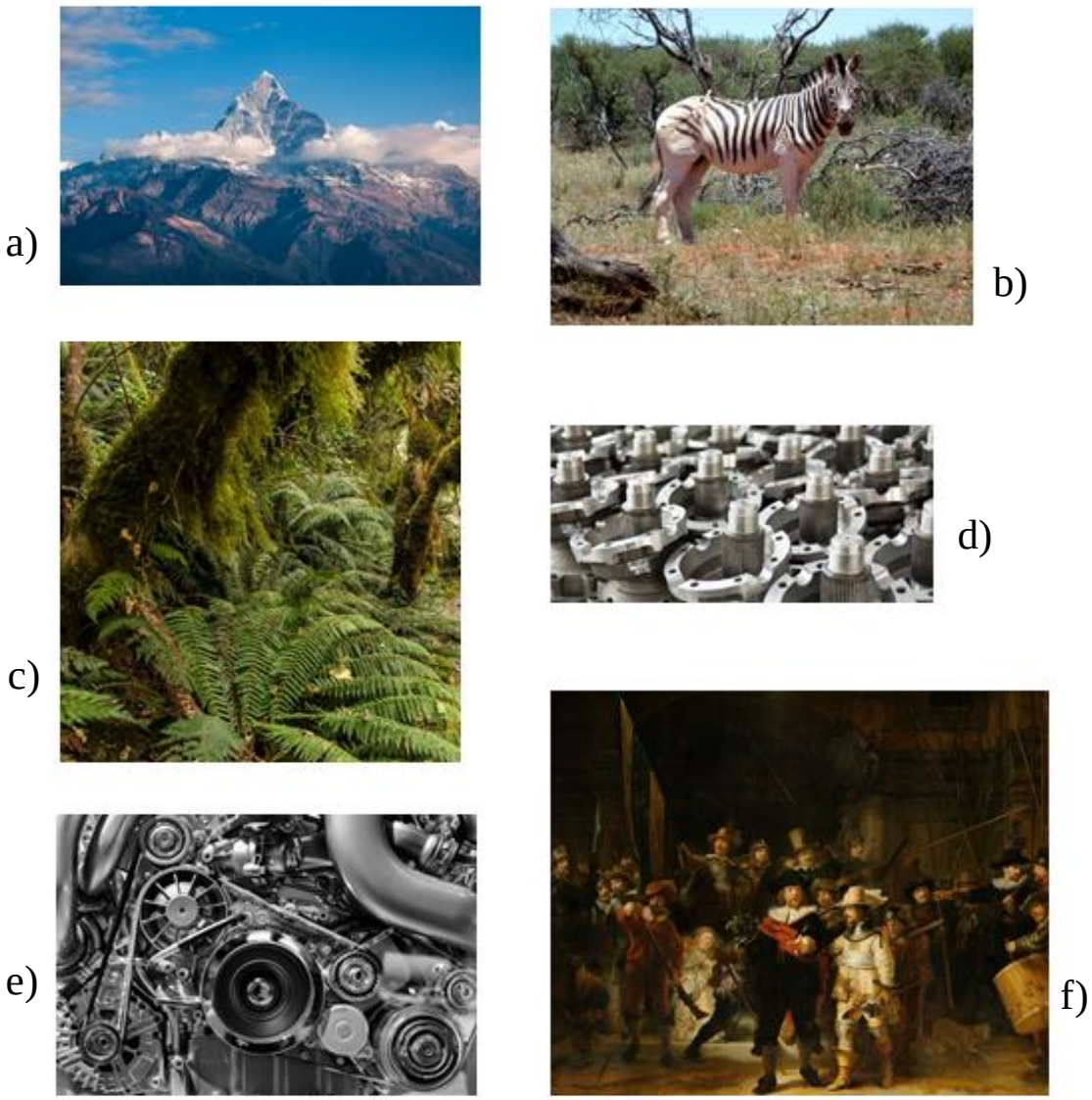


Fig. 4 Original six RGB images (a-f).

Table 1. Parameter *q* calculated for six images (a through f) modified by eight transformations (3.1 through 3.8).

| Fig. 4 Image | *q* for method described in section (subplot label) | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 3.1 | 3.2 | 3.3 | 3.4 | 3.5 | 3.6 | 3.7 | 3.8 |
| a | 0.2138 | 0.0017 | 1.8654 | 4.7523 | 0.0209 | 3.5789 | 1.2888 | 0.0018 |
| b | 0.8438 | 0.0014 | 2.6883 | 4.8623 | 0.0779 | 3.197 | 1.4423 | 0.0065 |
| c | 0.7399 | 2.65E-04 | 1.475 | 4.4791 | 0.0793 | 3.0354 | 1.4136 | 0.0066 |
| d | 0.5803 | 0.0054 | 3.8729 | 6.285 | 0.0978 | 2.8716 | 0.9473 | 0.0074 |
| e | 0.5746 | 0.0013 | 2.2391 | 5.8343 | 0.0533 | 3.0551 | 1.1334 | 0.0061 |
| f | 0.5313 | 2.93E-04 | 1.1302 | 1.3434 | 0.0254 | 2.7866 | 0.8098 | 0.0041 |

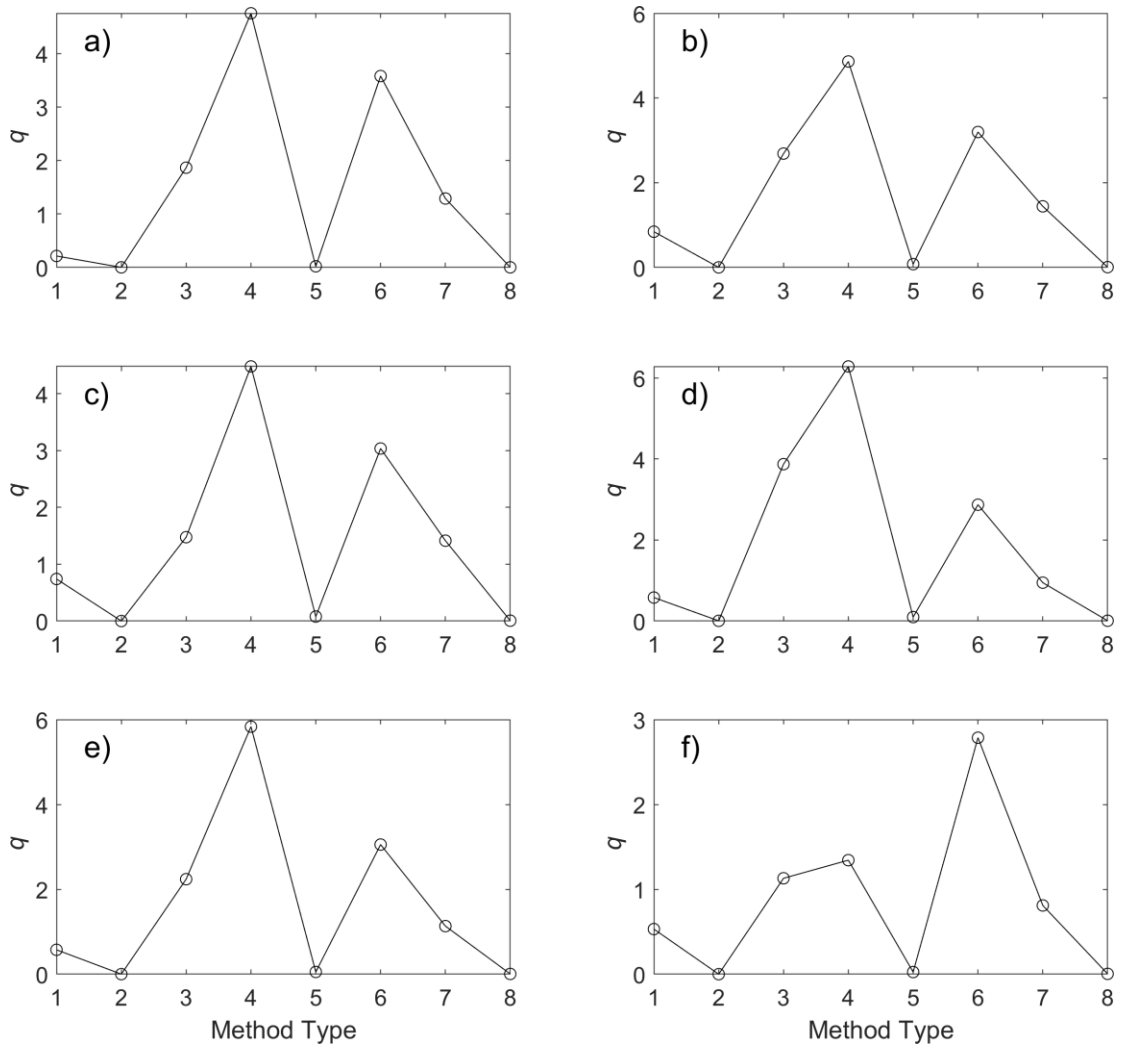


Fig. 5 Numerical results from Table 1 presented on graphs: parameter *q* for six images shown in Fig. 4(a-f) and modified by eight transformations.

Table 2. Image size, normalized *Q*, and image entropy *E*.

| Image | a | b | c | d | e | f |
|---|---|---|---|---|---|---|
| Size (pixels) | 500 x 333 | 1,562 x 1,172 | 2,390 x 2,500 | 329 x 153 | 1,912 x 1,272 | 2,765 x 2,250 |
| *Q* | 0.0446 | 0.1733 | 0.1651 | 0.0916 | 0.0983 | 0.3954 |
| *E* | 7.3311 | 7.6805 | 7.4418 | 7.9019 | 7.7090 | 6.4867 |

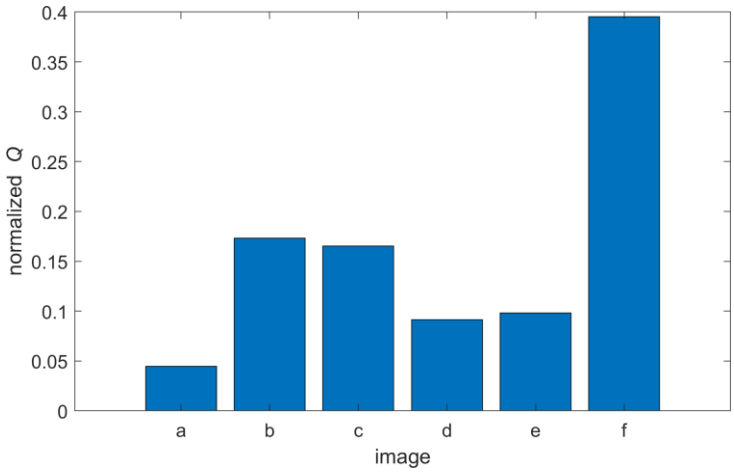


Fig. 6 Numerical results from Table 2: normalized parameter *Q* for six images.

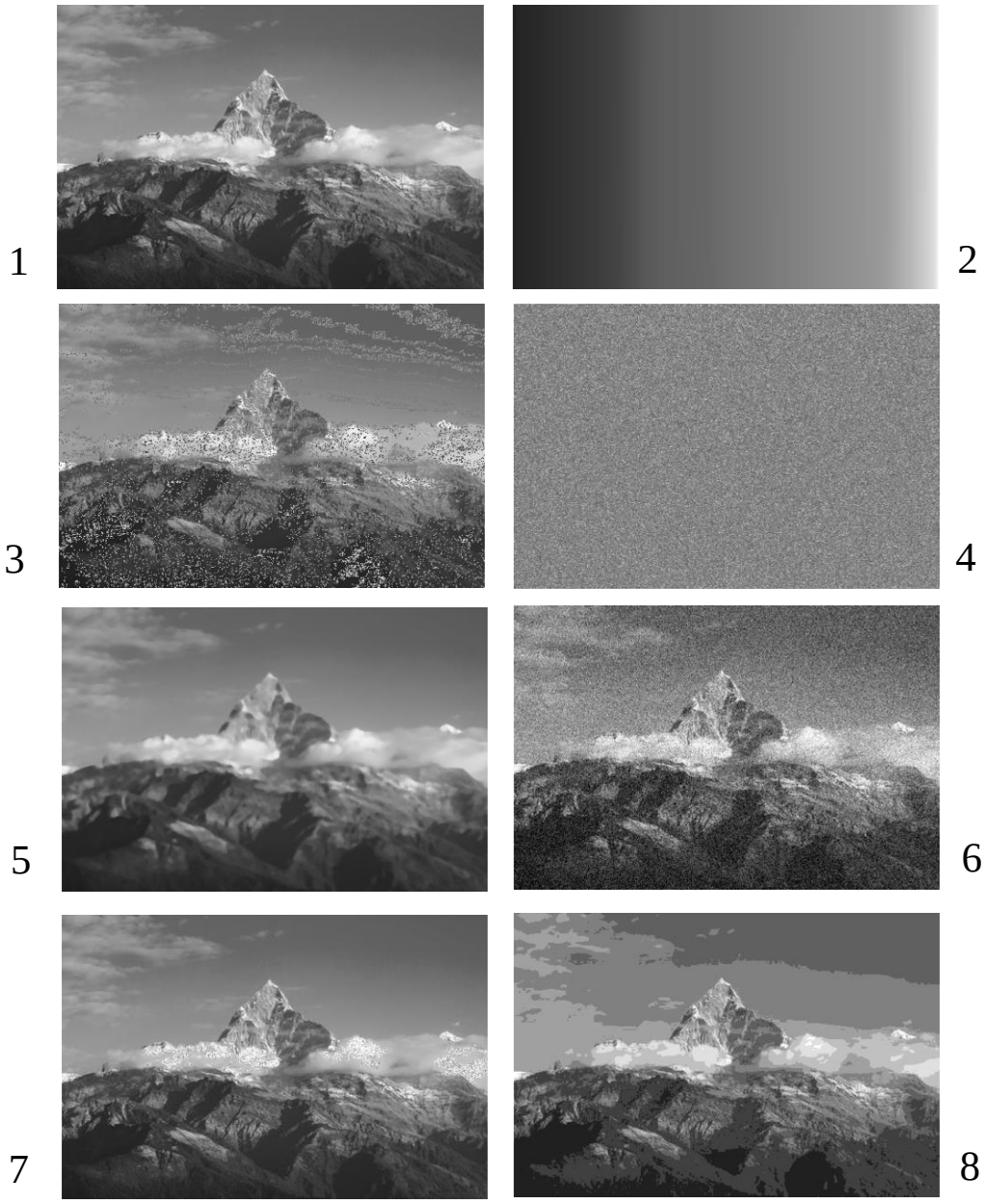


Fig. 7 Grayscale image obtained from RGB image shown in Fig. 4a and modified by the eight transformations described in Section 3.

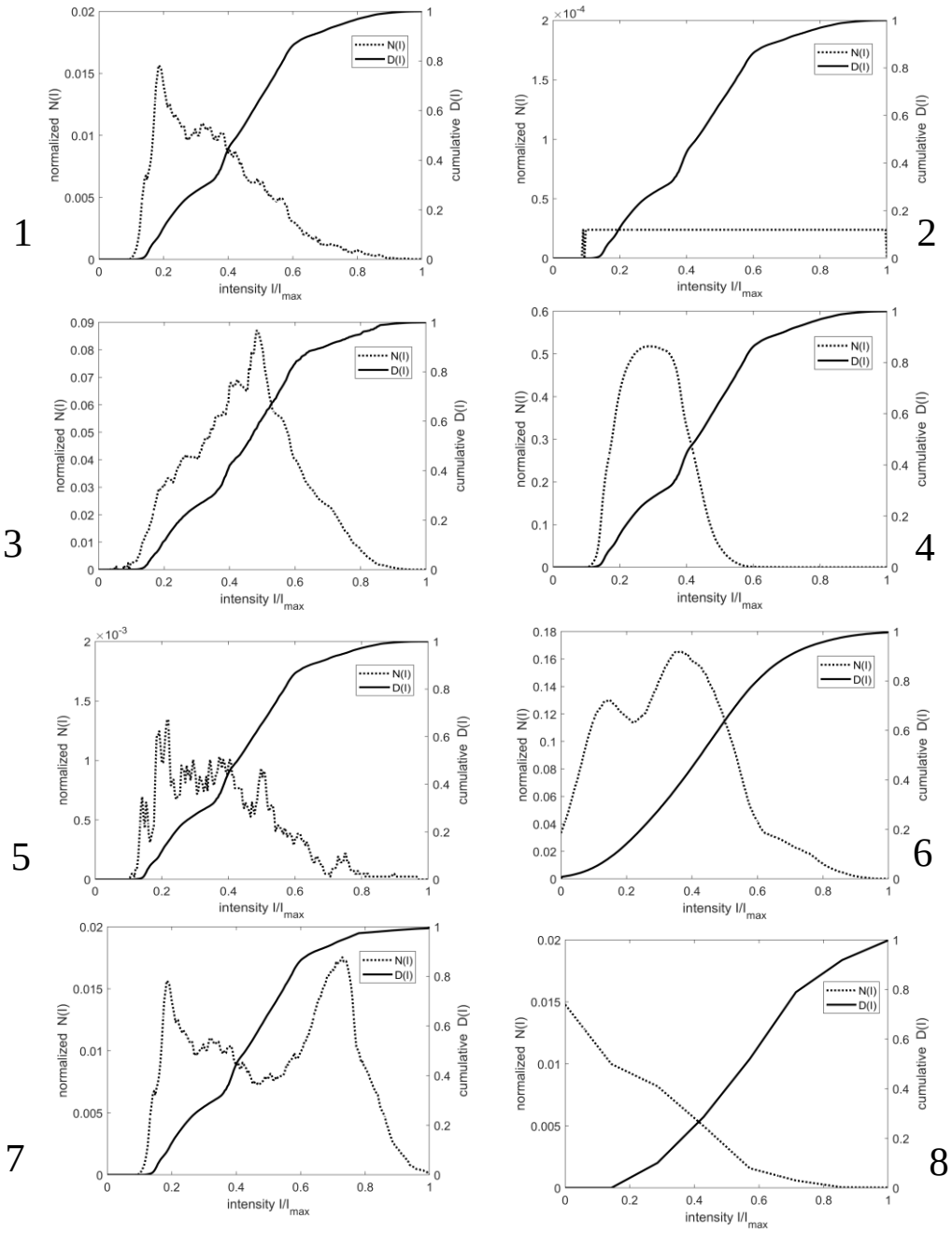


Fig. 8 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 7. Note change in scale for $N(I)$ on left axis in different subplots.

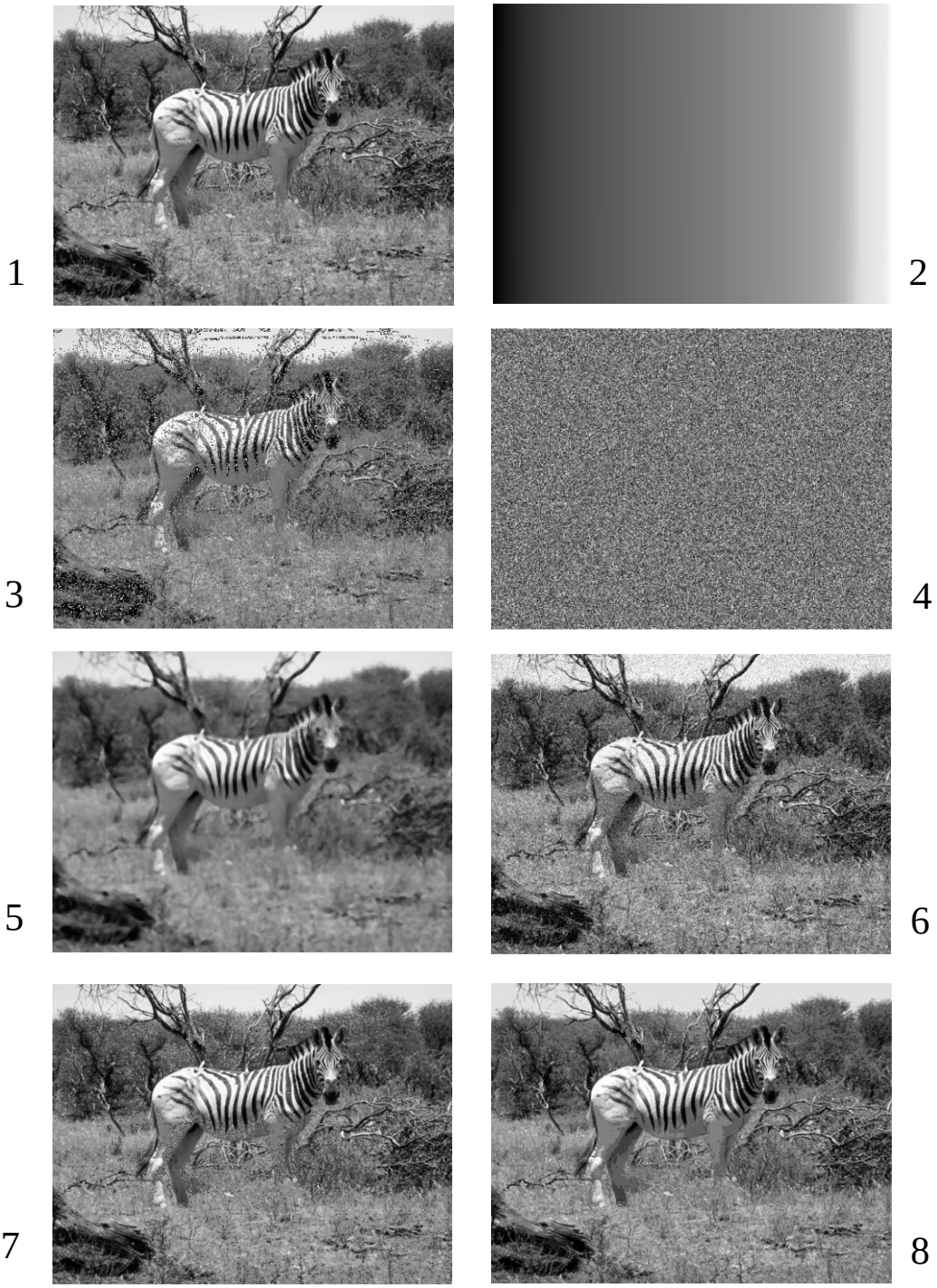


Fig. 9 Grayscale image obtained from RGB image shown in Fig. 4b and modified by the eight transformations described in Section 3.

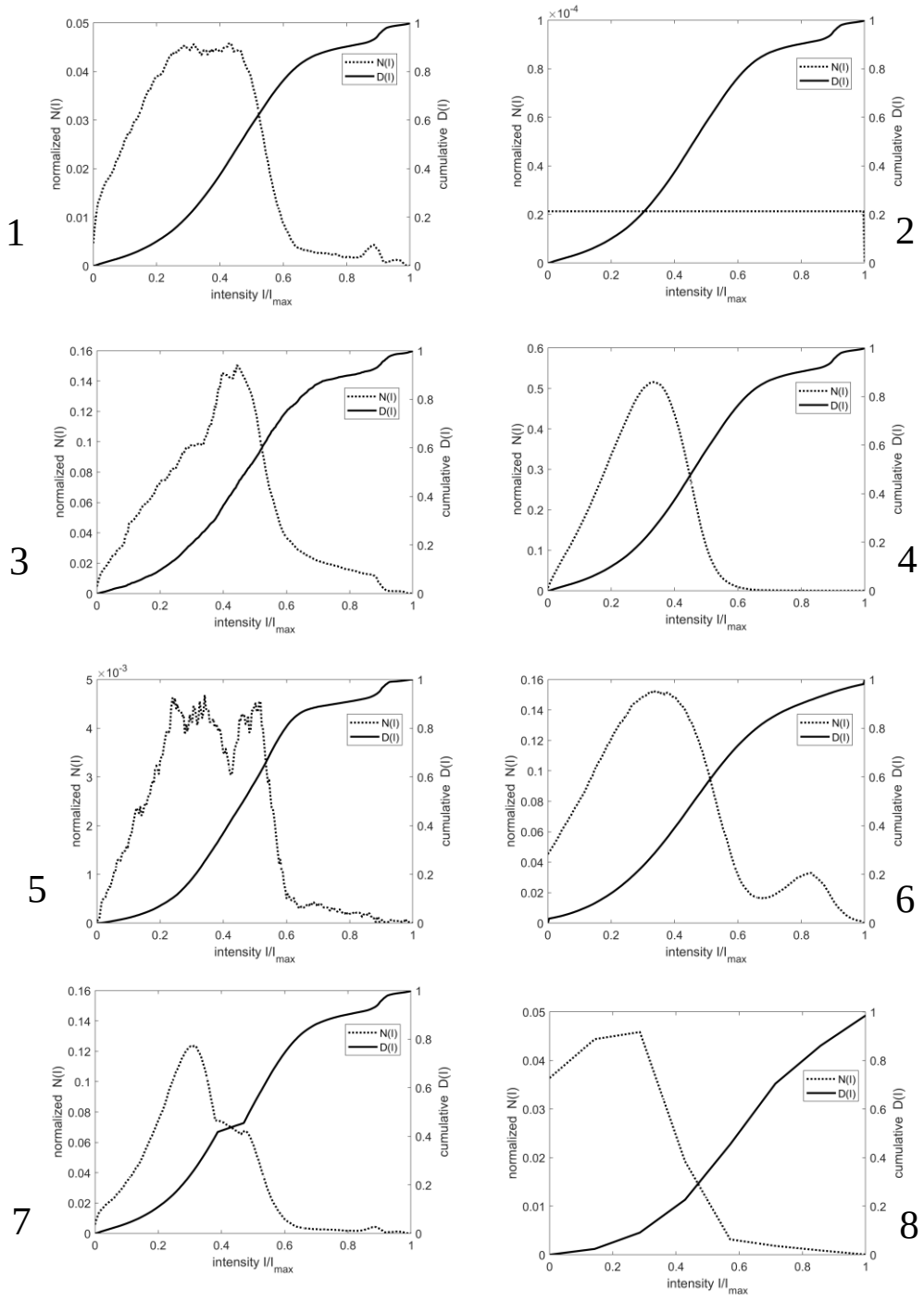


Fig. 10 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 9. Note change in scale for $N(I)$ on left axis in different subplots.

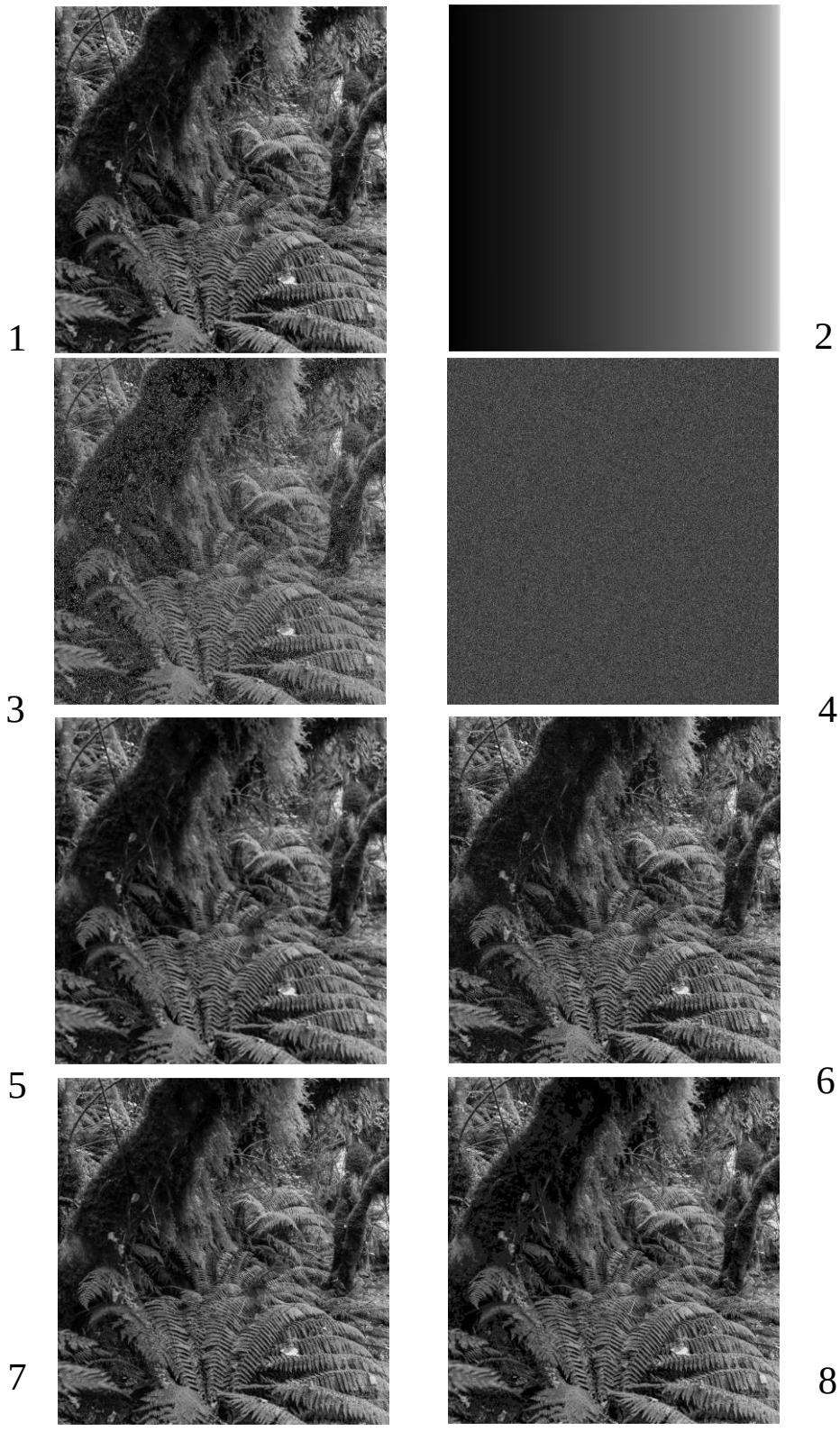


Fig. 11 Grayscale images from RGB image shown in Fig. 4c and modified by the eight transformations described in Section 3.

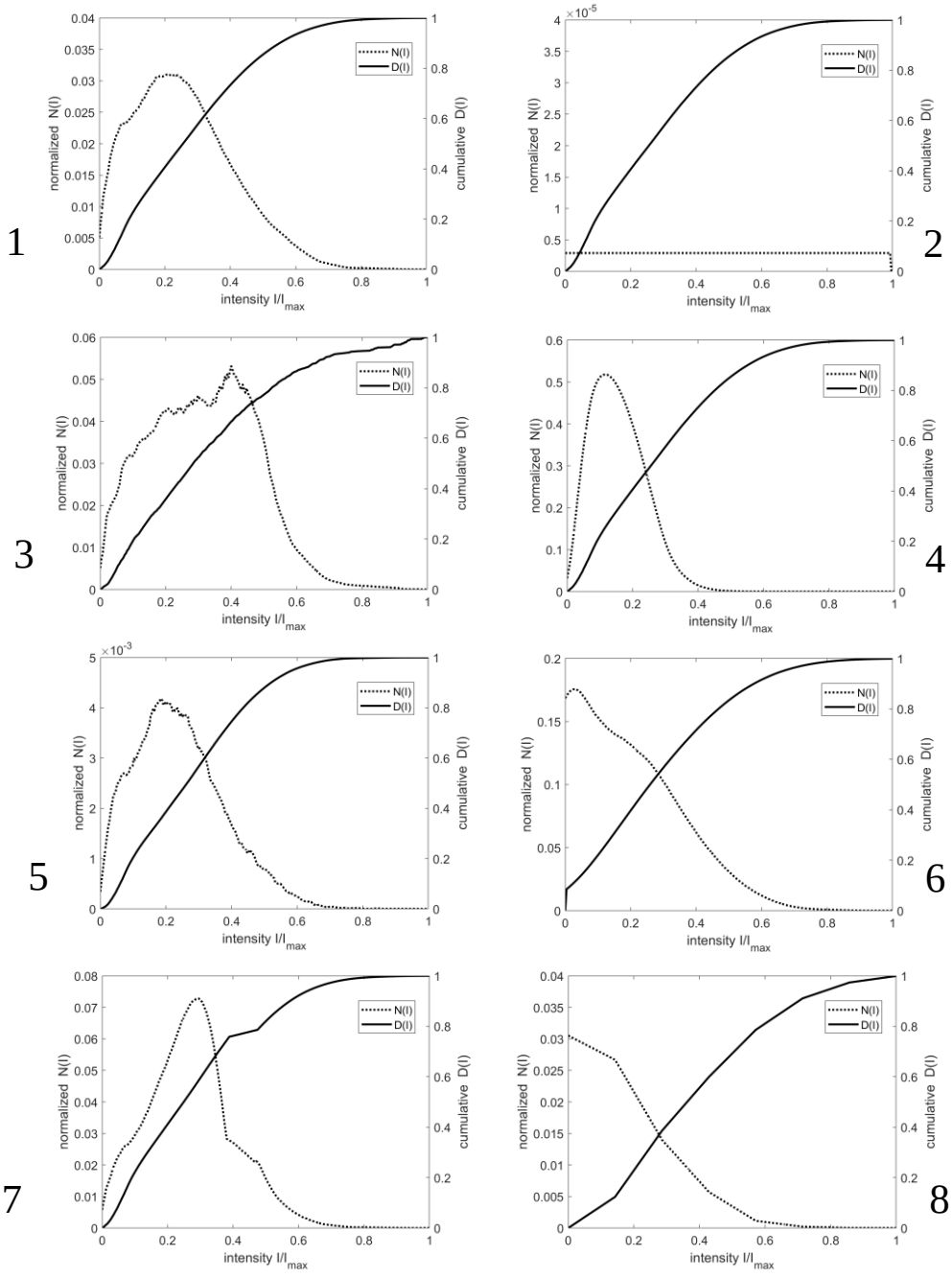


Fig. 12 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 11. Note change in scale for $N(I)$ on left axis in different subplots.

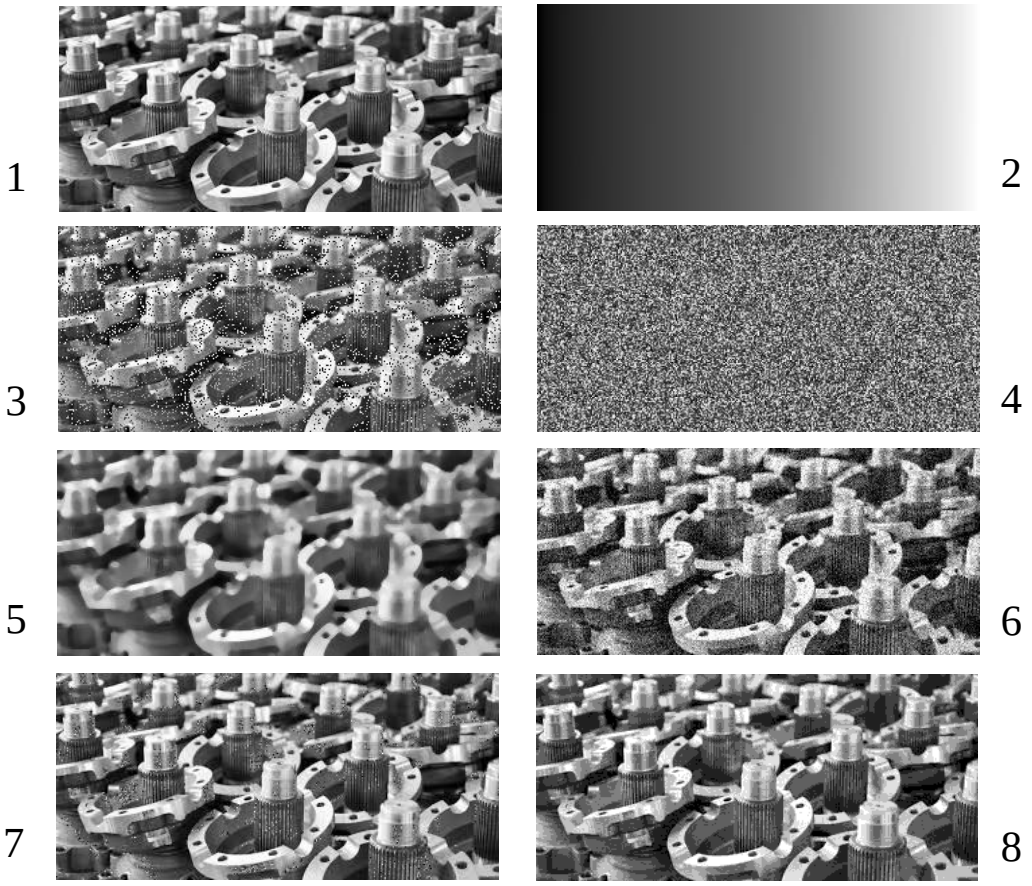


Fig. 13 Grayscale image obtained from RGB image shown in Fig. 4d and modified by the eight transformations described in Section 3.

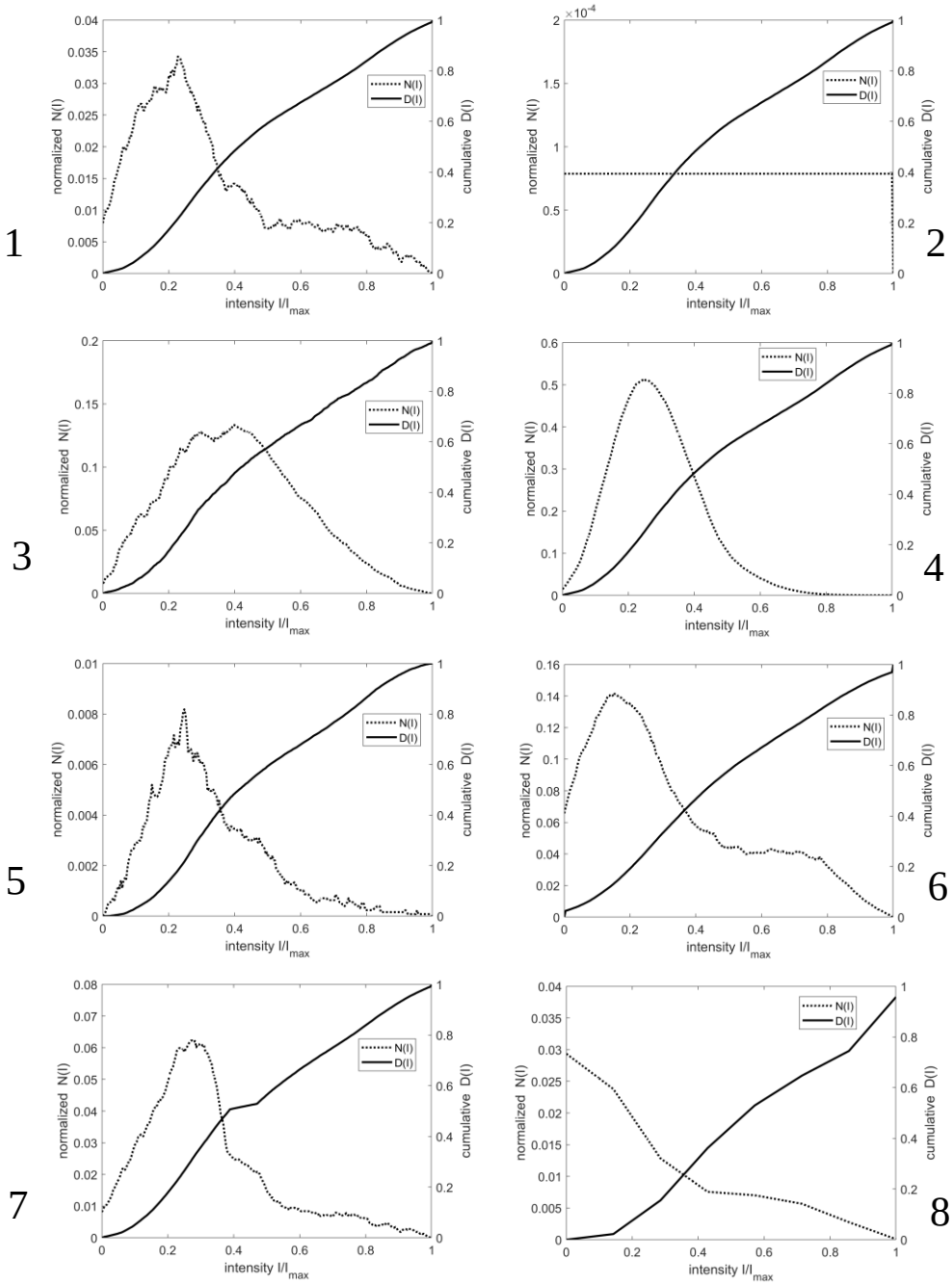


Fig. 14 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 13. Note change in scale for $N(I)$ on left axis in different subplots.

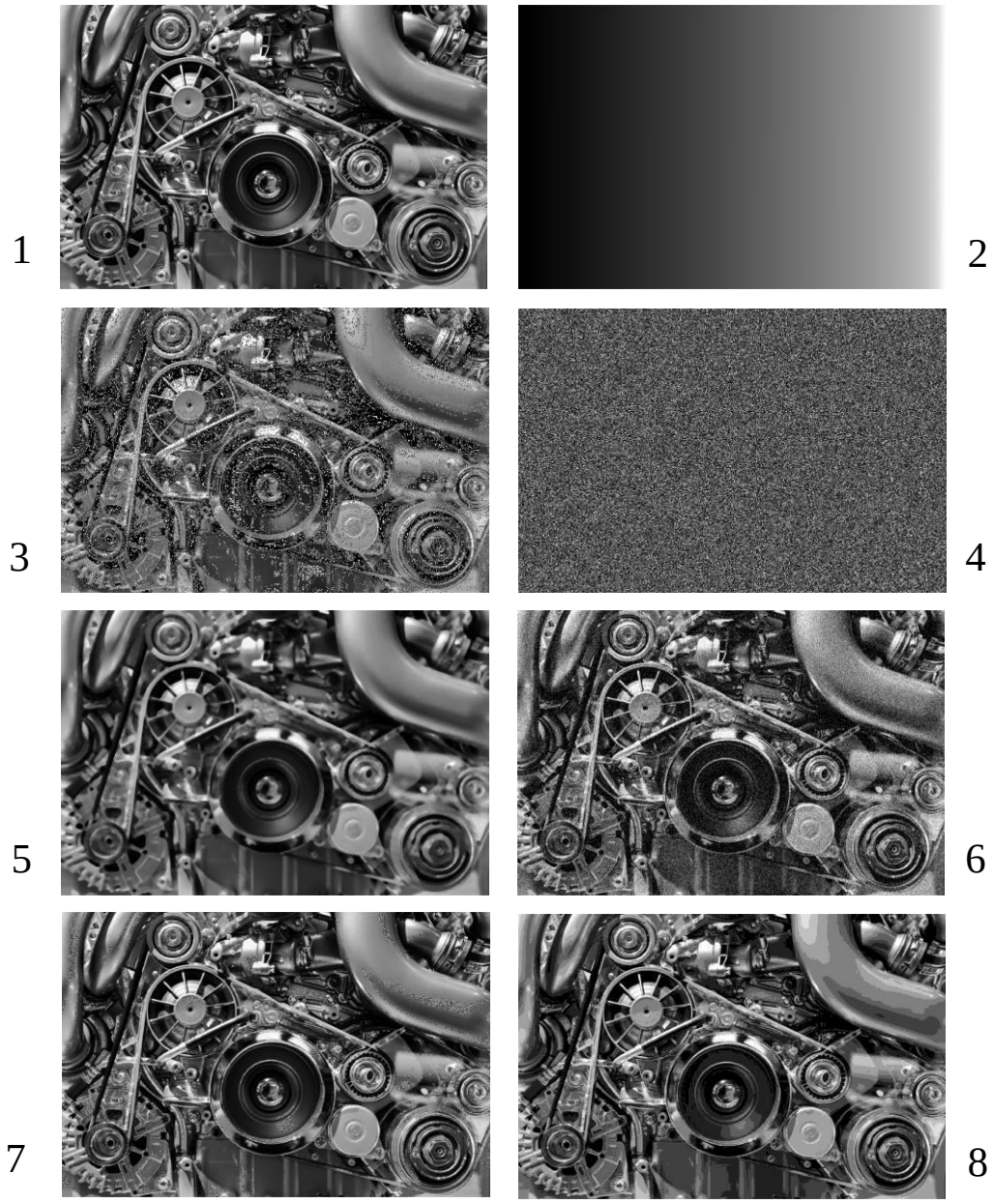


Fig. 15 Grayscale image obtained from RGB image shown in Fig. 4e and modified by the eight transformations described in Section 3.

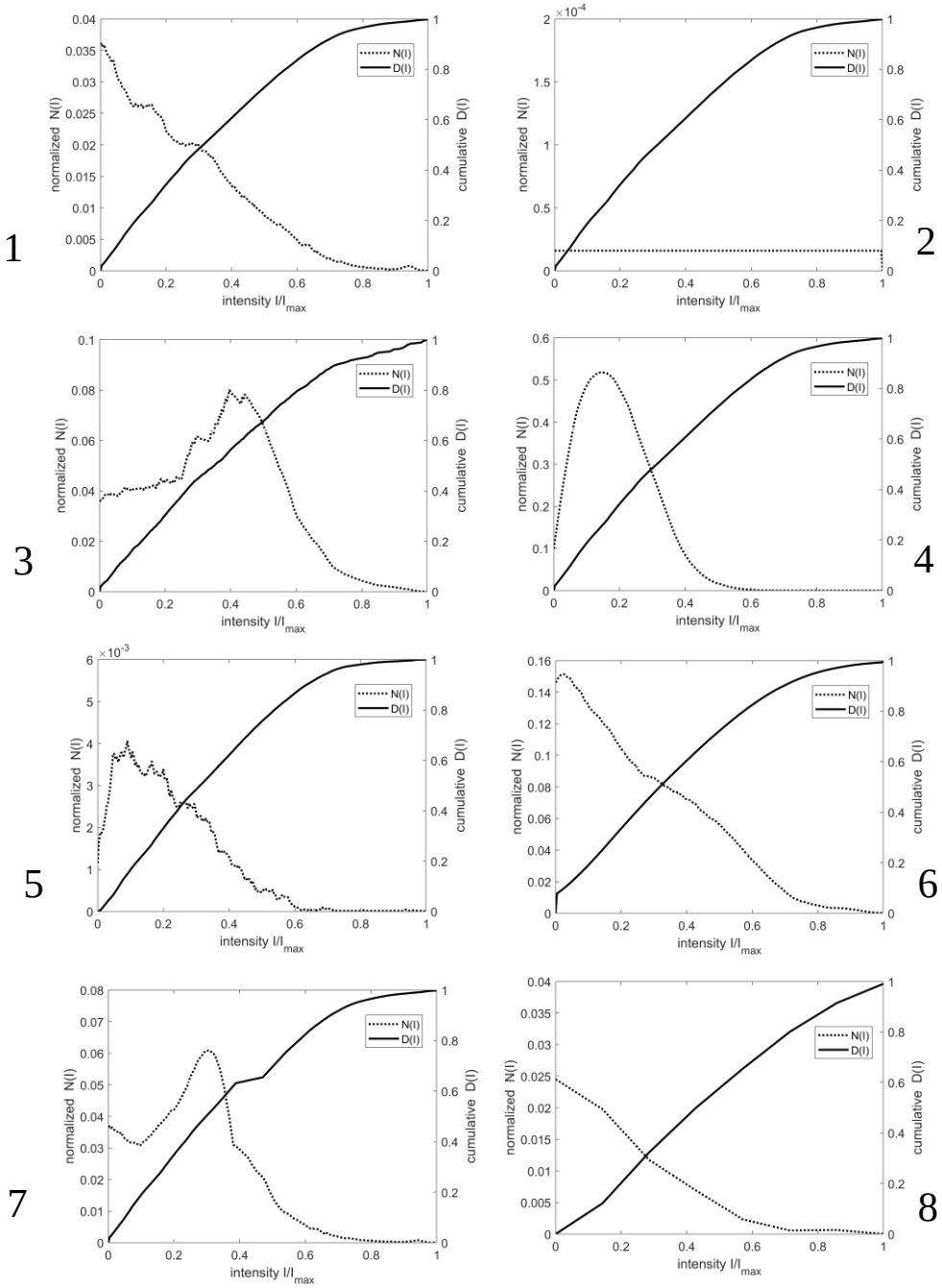


Fig. 16 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 15. Note change in scale for $N(I)$ on left axis in different subplots.

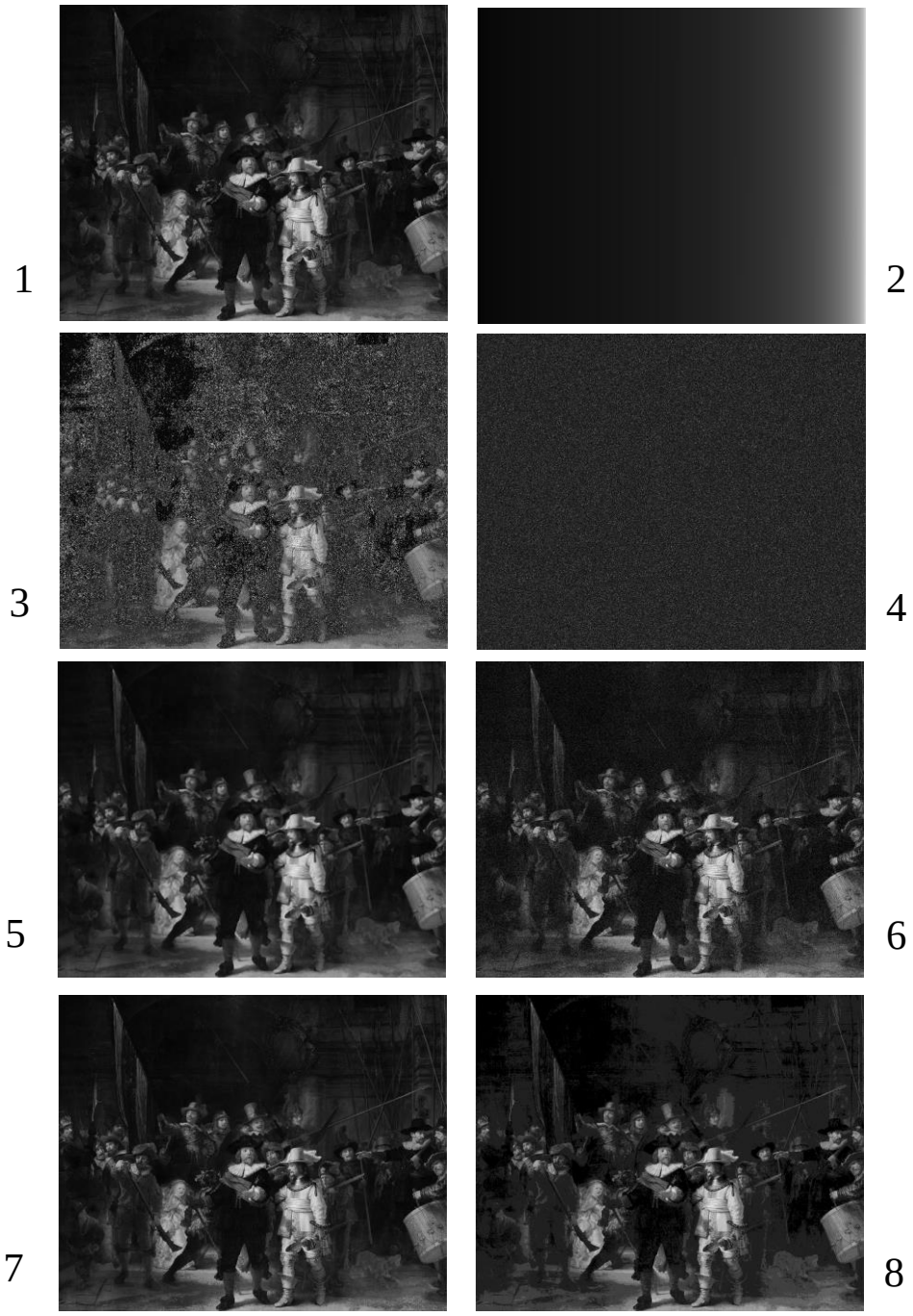


Fig. 17 Grayscale image obtained from RGB image shown in Fig. 4f and modified by the eight transformations described in Section 3.

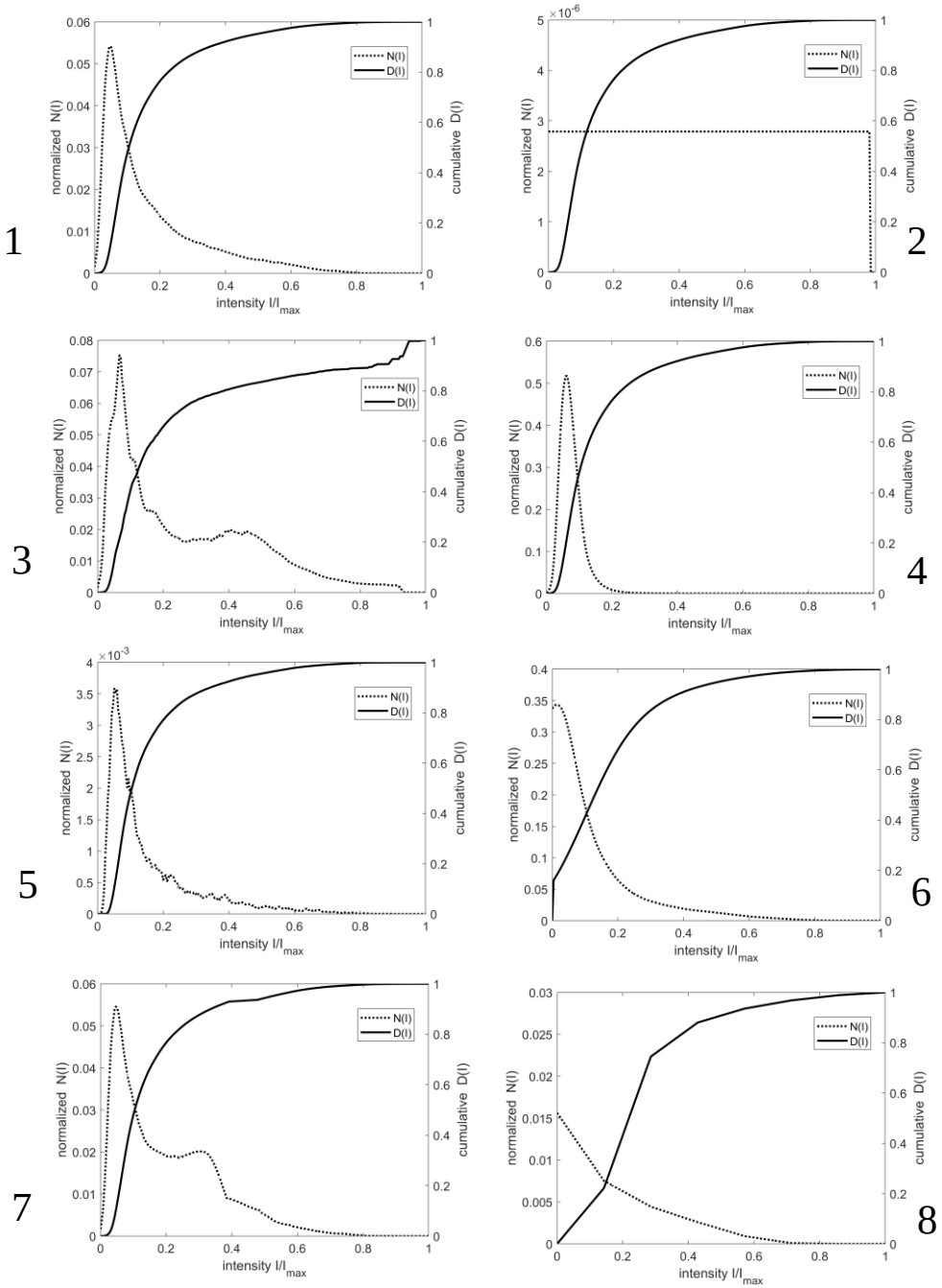


Fig. 18 Plots of $N(I)$ and $D(I)$ for eight grayscale images shown in Fig. 17. Note change in scale for $N(I)$ on left axis in different subplots.

## 5. Results for depth images

Two rectangular planar targets were scanned with two different sensors; each scan resulted in a depth image (A and B). Scanning conditions were such that the target was parallel to the

sensor's optical plane and a depth recorded at a given pixel was the shortest distance between a point on the target and the sensor's optical focal plane. In Fig. 19, depth histograms for both images are shown.

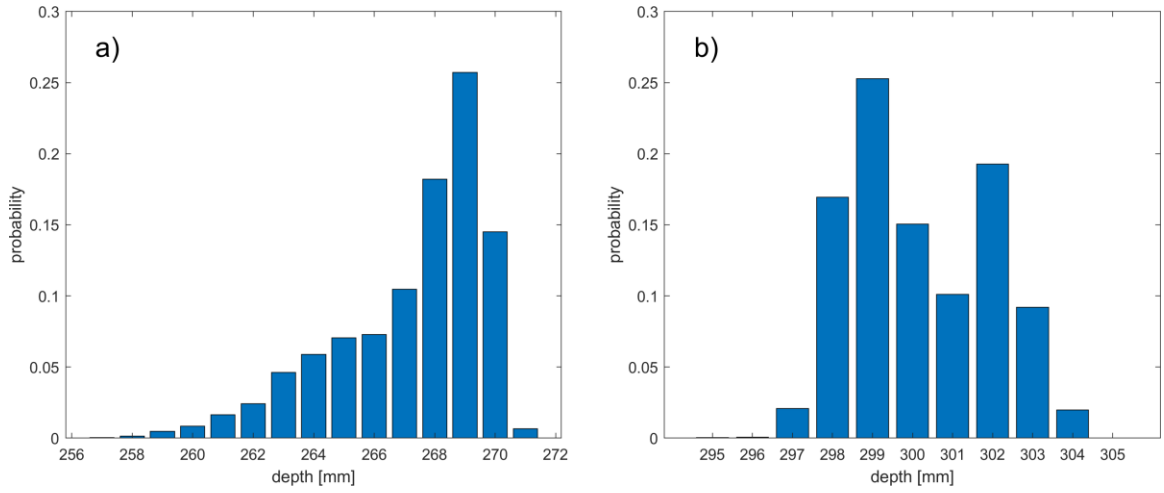


Fig. 19 Histogram of depths for data acquired with two different range cameras: a) A; b) B.

Since the acquired depths varied in a much smaller range than a typical range of grayscale images (256), parameter $q$ for depth images was calculated using only three methods: 1) identity transformation described in section 3.1; 2) grouping locations of pixels with the same depth (section 3.2), which yielded the smallest $q_{min}$; and 3) random reshuffling of pixel locations (section 3.4), which effectively yielded the largest $q_{max}$. Then, for both depth images, the normalized parameter $Q$ defined in Eq.(6) was calculated. Summary of the results (parameters $q$ and $Q$) are provided in Table 3. Both images processed by the three methods are shown in Fig. 20 and 21. The corresponding graphs of $N(I)$ and $D(I)$ are shown in Fig. 22 and 23, respectively, where $I$ is a discrete value of depth measured by the sensor.

Table 3. Results of processing depth images.

| Depth Image | Size (pixels) | Normalized $Q$ | $q$ for method described in section (subplot label) | | |
|---|---|---|---|---|---|
| | | | 3.1 | 3.2 | 3.4 |
| A | 720 x 1,280 | 1.75E-03 | 5.63E-04 | 1.03E-05 | 0.3153 |
| B | 240 x 385 | 3.06E-03 | 6.56E-04 | 1.26E-04 | 0.1735 |

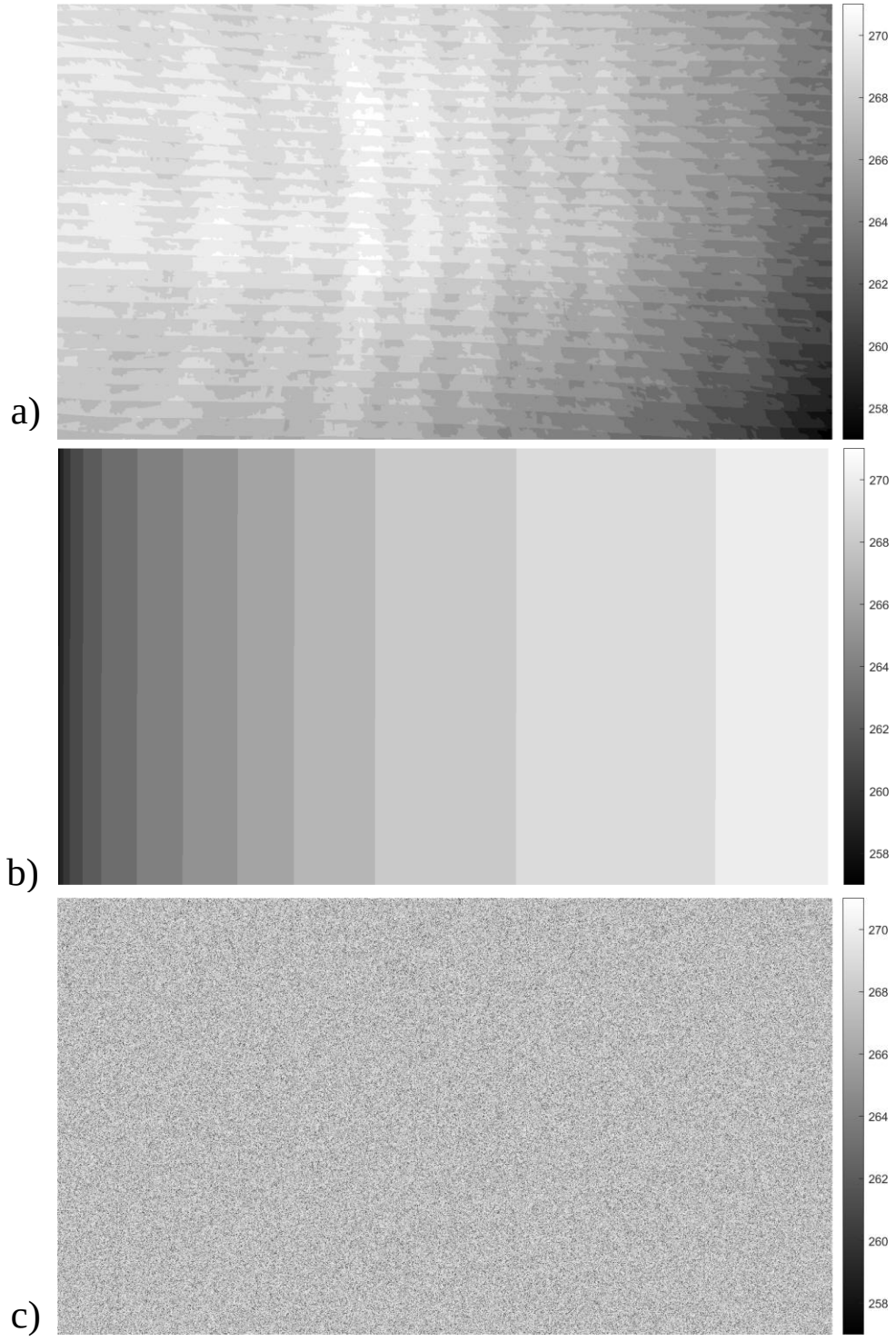


Fig. 20 Depth image acquired with range camera A: a) processed by the identity transformation described in 3.1 (original image); b) modified by transformation grouping the same intensities (yields $q_{min}$) described in 3.2; c) modified by random reshuffling (yields $q_{max}$) described in 3.4. The units of the grayscale intensity scale to the right of each image are in mm.

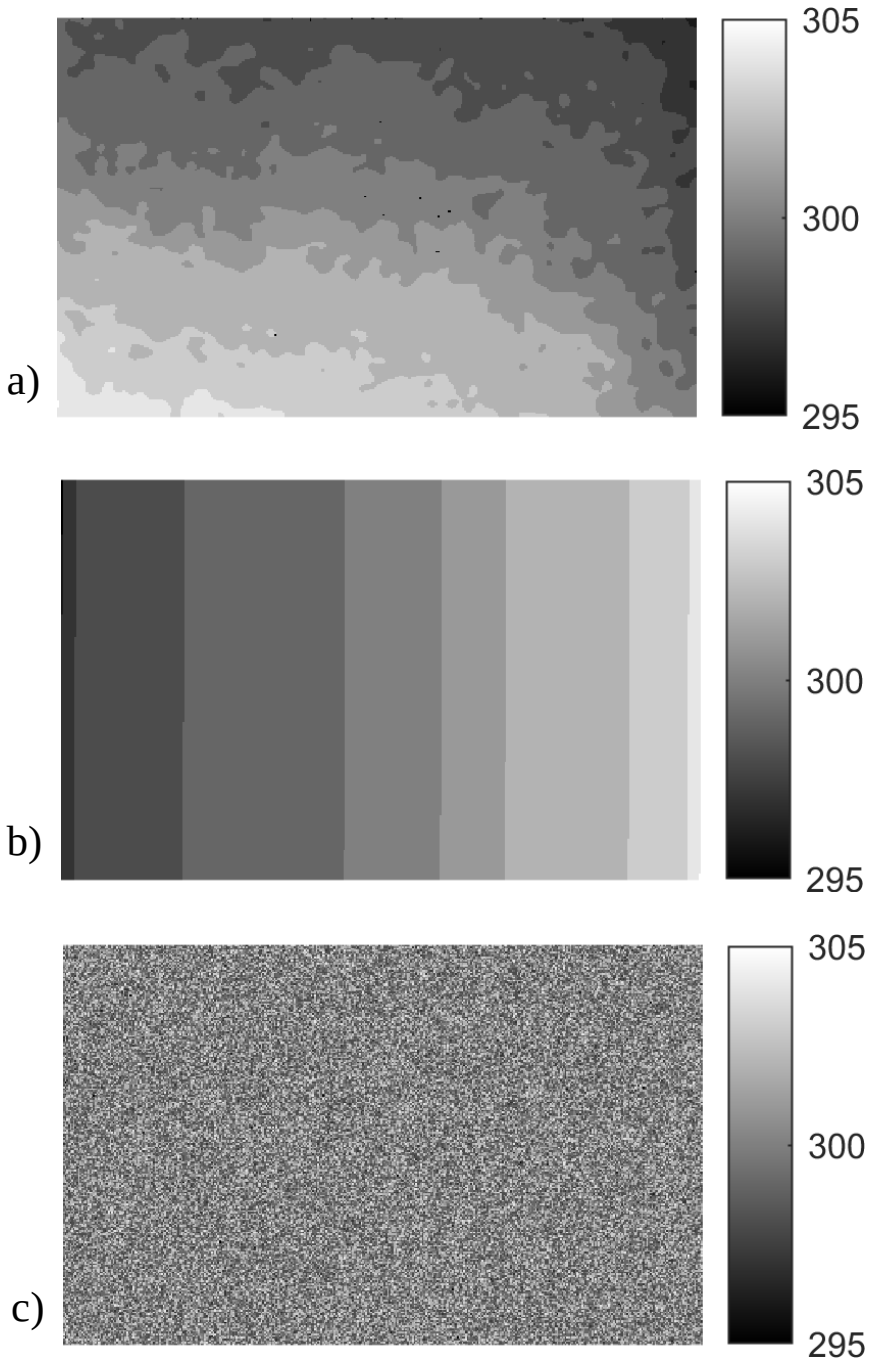


Fig. 21 Depth image acquired with range camera B: a) processed by the identity transformation described in 3.1 (original image); b) modified by transformation grouping the same intensities (yields $q_{min}$) described in 3.2; c) modified by random reshuffling (yields $q_{max}$) described in 3.4. The units of the grayscale intensity scale to the right of each image are in mm.

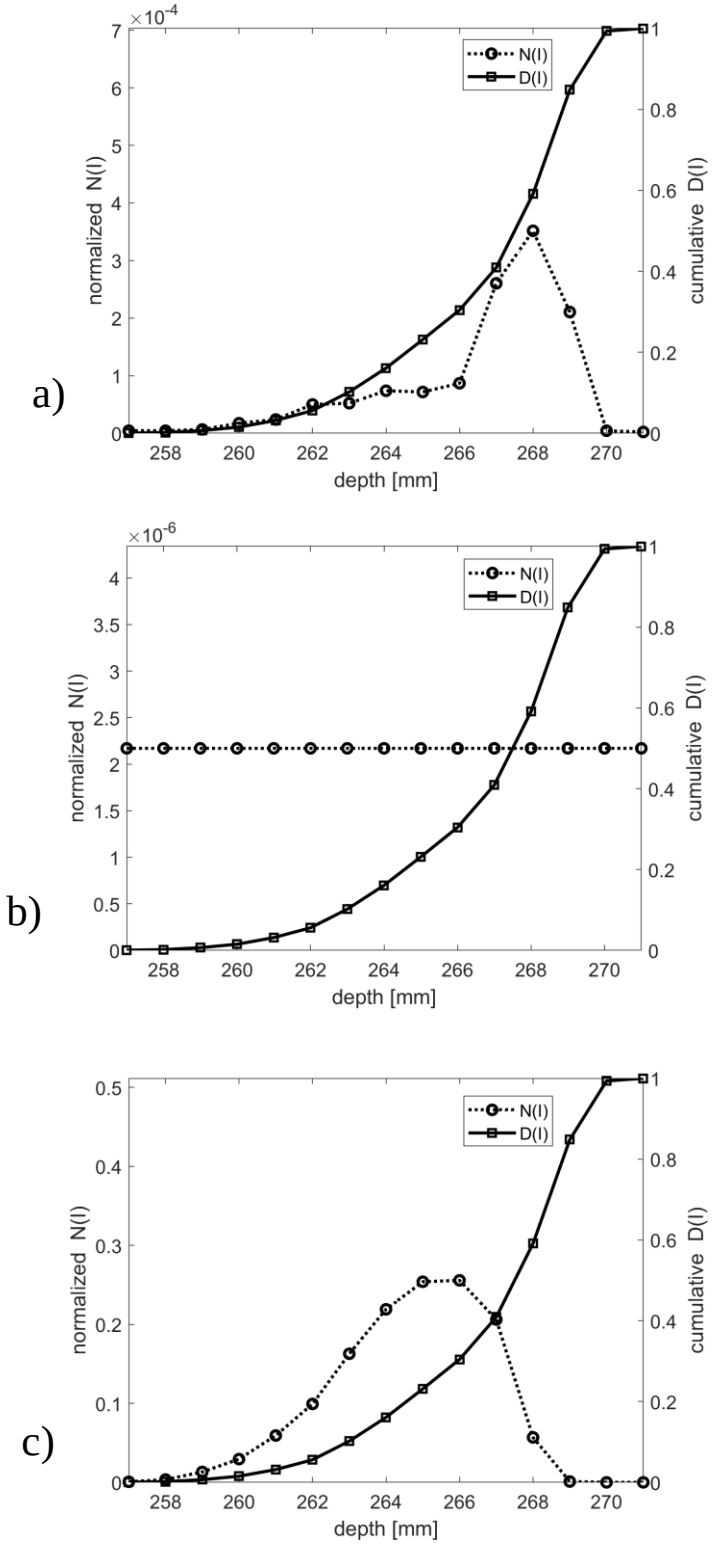


Fig. 22 Plots of $N(I)$ and $D(I)$ for three depth images shown in Fig. 20(a-c). Here, $I$ indicates depth in mm. Note the change in scale for $N(I)$ on the left axis in the different subplots.

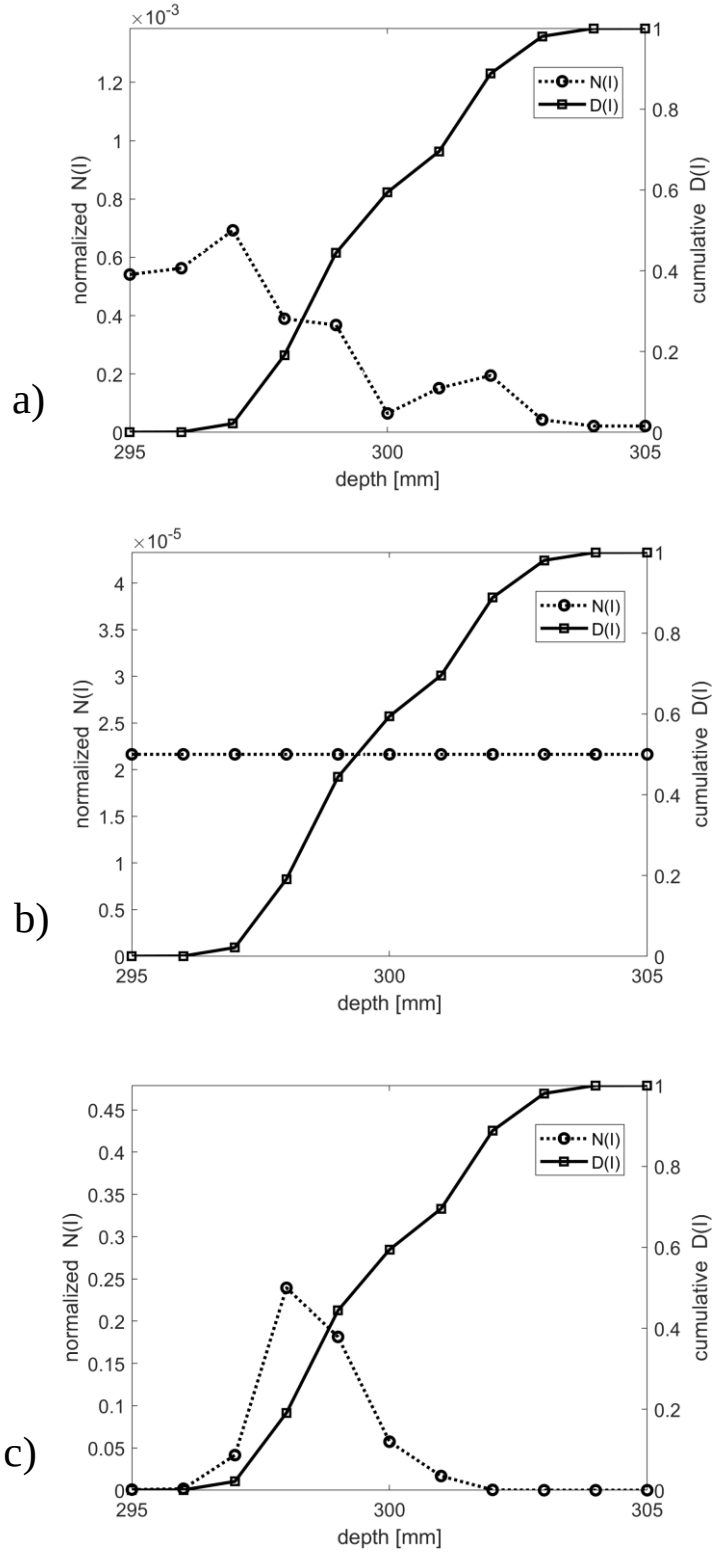


Fig. 23 Plots of $N(I)$ and $D(I)$ for three depth images shown in Fig. 21(a-c). Here, $I$ indicates depth in mm. Note the change in scale for $N(I)$ on the left axis in the different subplots.

## 6. Discussion

The normalized parameter $Q$ (derived from parameters $q$, $q_{min}$, and $q_{max}$) introduced in this report is intended to provide a quantitative characterization of the difficulty of segmentation of 2D images. Segmenting an image with many small clusters (and the need to find a starting seed for each cluster) is more cumbersome and time consuming than segmenting an image with fewer larger clusters. The process of segmentation is sensitive to the profile $D(I)$ of a recorded signal $I$ (intensity of light, or depth in the examples presented in this report) as well as the spatial distribution of pixels with the same signal. Any meaningful analysis of image content starts with determining clusters of connected pixels which share a common criterion, e.g., all pixels in a single cluster have signal below a threshold $I$. It is the number and, consequently, the sizes of the clusters that make the segmentation task simple or complex and, ultimately, may cause a semantic analysis of the image content to be easy or difficult. The degree of difficulty of such analyses depends not only on the image quality (e.g., clear, noisy, blurry, or sharp picture), but also on the content itself (e.g., whether the physical scene is cluttered or the object of interest occupies a small portion of the scene).

Images processed according to the procedures described in Sec. 3.2 and 3.4 are shown in subplots 2) and 4) in Figs. 7, 9, 11, 13, 15, and 17. While the meaningful content of the original images are totally lost for each of these examples, the resulting images are important as they constitute the boundary images that yield the smallest and the largest parameters $q_{min}$ and $q_{max}$, respectively, for all images of the same size and intensity profile $D(I)$. From comparing images in subplots 2) with those in subplots 4), it is clear that segmentation of the first type of images is much easier. By contrast, semantic analysis of images in subplots 4) is extremely difficult due to a large number of small clusters.

Out of eight processing procedures described in Sec. 3.1-3.8, only three (Sec. 3.1, 3.2 and 3.4) preserve the signal profile $D(I)$. The remaining five procedures affect $D(I)$ to various extents. All eight procedures have much stronger effects on the distribution of the number of clusters $N(I)$, as demonstrated by the graphs in Figs. 8, 10, 12, 14, 16, and 18. Changes in both $D(I)$ and $N(I)$ result in different $q$ values computed by all eight procedures, as shown in Fig. 5. The addition of different types of noise to the original images (methods 3, 6, and 7 in Fig.5) led to an increase in $q$ value, while median filtering and decreases in intensity range (methods 5 and 8 in Fig. 5) caused reductions in the values of parameter $q$ for all six images shown in Fig. 4 (see also Table 1). This pattern is expected because the addition of noise makes segmentation more difficult as noise impacts the acceptance criteria for individual pixels. This leads to a split of the original clusters of connected pixels into smaller subclusters and increases the total number of clusters. On the other hand, smoothing the boundaries by the median filter or decreasing the color depth of the original images tends to decrease the number of clusters and makes the task of segmentation easier as fewer starting seeds need to be found.

Normalized parameters $Q$ for all six images (a-f) are shown in Fig.6 and in the bottom row of Table 1. Two possible use cases of the parameter $Q$ are discussed below.

### 6.1. Gauging variability of RGB images in largescale datasets

To effectively use machine learning (ML) methods for object recognition in RGB images, datasets with a large collection of images are used to train object recognition algorithms. *"How good"* the training set is—how well the algorithms perform on test cases—typically translates to "*how large*" the dataset is. However, the size of the dataset alone may be an insufficient metric. What is really needed is some metric of the variability of the images in the dataset.

The normalized parameter *Q* introduced in this report may be potentially useful for the purpose of determining the variability of images in a dataset. A distribution of the scalar parameter *Q* provides some insight into how hard or easy it is to segment images in the dataset. However, the segmentation process does not depend on the semantic content of the image and, therefore, building the training dataset with all relevant categories equally represented must be carefully monitored as the problem of nonuniform distribution of categories in large datasets cannot be addressed by *Q*.

It would be improper to expect that a single scalar parameter can fully characterize something as complex as a 2D image. Furthermore, segmentation is only a first step in the multistage process of object recognition. The value of *Q* does not necessarily indicate how easy or difficult it is to find the object in the image. For example, the image in Fig. 4b shows a zebra and has a larger *Q* (which indicates higher segmentation difficulty), but finding the zebra in this image may actually be easier than finding a screw in Fig. 4e, an image that is nominally easier to segment as it has a smaller *Q*. However, assuring that images included in the training dataset cover a wide spectrum of segmentation conditions should enhance the performance of ML algorithms on test images.

### 6.2. Gauging experimental conditions for testing depth cameras

Many sensors that produce 3D point clouds calculate coordinates of 3D points from a measured depth image. One of the most important metrics in evaluating the performance of such sensors is depth error, which is measured in carefully-designed experimental setups (e.g., a sufficiently flat target parallel to the focal plane of the sensor). Providing a qualitative assessment of the experimental conditions under which the performance of depth cameras are evaluated is not easy. It is obvious that the experimental conditions under which the depth images in Fig. 20a and Fig. 21a were acquired are far from optimal: in the ideal settings, each depth image should contain only one value. In reality, depth histograms shown in Fig. 19 exhibit a dispersion of depths which, additionally, are strongly correlated on the image plane. This strong clustering of pixels on depth images causes small values of the parameter *Q* shown in Table 3. If the resulting depth images were instead as in Fig. 20c and Fig. 21c (where no clustering of pixels with the same depth occurs) then a corresponding parameter *Q* would be close to one. Thus, parameter *Q* may be useful to quantitively evaluate the experimental conditions needed for testing depth cameras.

## 7. Final comments

What constitutes a well-balanced dataset of images remains to be investigated. In the context of semantic categories represented in a well-balanced dataset, this means that each category contains the same number of images included in a dataset. Extrapolating this interpretation to the parameter *Q*, one may similarly expect that a well *Q*-balanced dataset has a uniform distribution of *Q*. However, if the parameter *Q* gauges the difficulty of segmentation then one may argue that images with larger *Q* (i.e., more difficult to segment) should be more frequently represented in a dataset than images with small *Q*. Training models on more images that are difficult to segment, as opposed to images that are fairly easy to segment, could boost a training process and result in a model with better performance.

The problem of long tail distributions of categories as severe as in the LVIS dataset may be less troublesome for images in another domain. For example, parts used in assembly lines in manufacturing may be grouped in much fewer categories and it may be quite possible to collect a largescale dataset of images with all categories equally represented. However, images taken in industrial environments are known to exhibit large variability (lighting conditions, reflections from shiny surfaces, etc.) and thus, parameter *Q* may be especially useful for building a representative dataset of such images for training ML algorithms.

## References


[1] T.-Y. Lin *et al.*, "Microsoft COCO: Common Objects in Context," *arXiv:1405.0312v3 [cs.CV]*, pp. 1-15, 21 Feb 2015.

[2] X. Zhu, C. Vondrick, D. Ramanan, and C. Fowlkes, "Do We Need More Training Data or Better Models for Object Detection?," presented at the The British Machine Vision Conf. BMVC'12, Surrey, UK, 2012.

[3] A. Gupta, P. Dollar, and R. Girshick, "LVIS: A Dataset for Large Vocabulary Instance Segmentation " *arXiv:1908.03195v2 [cs.CV]*,

[4] *WordNet - A Lexical Database for English*. Available: https://wordnet.princeton.edu/

[5] D. M. Chandler, "Seven Challenges in Image Quality Assessment: Past, Present, and Future Research," *ISRN Signal Processing*, vol. 2013,

[6] R. Dosselmann and X. D. Yang, "Existing and emerging image quality metrics," in *Canadian Conference on Electrical and Computer Engineering*,, 2005, pp. 1906 - 1913.

[7] A. Mason *et al.*, "Comparison of Objective Image Quality Metrics to Expert Radiologists' Scoring of Diagnostic Quality of MR Images," *IEEE Trans. Medical Imaging*, vol. 39, no. 4, pp. 1064 - 1072, 2020.

[8] N. R. Pal and S. K. Pal, "A review on image segmentation techniques," *Pattern Recognition*, vol. 26, no. 9, pp. 1277-1294, 1993.

[9] H. P. Narkhede, "Review of Image Segmentation Techniques," *Int. J. of Science and Modern Engineering*, vol. 1, no. 8, pp. 54 - 61, 2013.

[10] H. Zhang, J. E. Fritts, and S. A. Goldman, "Entropy-based Objective Evaluation Method for Image Segmentation," in *Electronic Imaging*, San Jose, USA, 2004, vol. SPIE 5307, pp. 38 - 49.

[11] ITU. *Recommendation ITU-R BT.601-7*. Available: https://www.itu.int/rec/R-REC-BT.601-7-201103-I/en
[12] M. J. Kumar, G. R. Kumar, and R. V. K. Reddy, "Review on Image Segmentation Techniques," *Int. J. of Scientific Research Engineering and Technology*, vol. 3, no. 6, pp. 992 - 997, 2014.
[13] D. Kaur and Y. Kaur, "Various Image SegmentationTechniques: A Review," *Int. J. of Computer Science and Mobile Computing*, vol. 3, no. 5, pp. 809 - 814, 2014.