Hexagonal Image Processing for Computer Vision With Hexnet: A Hexagonal Image Processing Data Set and Generator

TOBIAS SCHLOSSER[1] , **MICHAEL FRIEDRICH**[1] , **TRIXY MEYER**[1] , **MAXIMILIAN EIBL**[2] , and **DANNY KOWERKO**[1]

[1]Junior Professorship of Media Computing, Chemnitz University of Technology, 09107 Chemnitz, Germany (e-mail: firstname.lastname@cs.tu-chemnitz.de)
[2]Professorship of Media Informatics, Chemnitz University of Technology, 09107 Chemnitz, Germany (e-mail: firstname.lastname@cs.tu-chemnitz.de)

Corresponding author: Tobias Schlosser (e-mail: tobias.schlosser@cs.tu-chemnitz.de).

ABSTRACT In the domains of image processing and computer vision, the exploration of hexagonal image processing systems has emerged as a fundamentally innovative yet nascent methodology that is motivated by the occurrence of hexagonal structures in the human visual perception system and nature itself. However, despite the possible benefits of hexagonal over conventional square approaches for image processing systems – which commonly utilize square pixels – no known publicly available hexagonal image data sets exist that would enable the evaluation of hexagonal approaches that have been developed within image processing and computer vision for tasks such as object detection and classification. For this purpose, this contribution proposes a foundation for hexagonal image data sets and their development: The *Hexnet Hexagonal Image Processing Data Set* (short *Hexnet Dataset*), which is based on The *Hexagonal Image Processing Framework Hexnet* (*Hexnet Framework*). As a baseline, three data subsets are introduced: (i) *geometric primitives* for the evaluation of hexagonal structures, (ii) *astronomical image processing*, in which the descriptions of sensory elements of hexagonal telescope arrays have been leveraged for the detection and classification of synthesized atmospheric events, and (iii) *conventional image data sets*, which provides hexagonally transformed versions of commonly evaluated square imagery.

I. INTRODUCTION AND MOTIVATION

WITH the rise of artificial neural networks (ANN) [1]–[3], recent advancements have markedly elevated the significance of image processing and computer vision in various domains [4]. The increasing complexity of novel problems, the emergence of innovative application areas, and the expansion of data sets necessitates the development of novel models and procedures [5]–[7]. Nowadays, approaches such as deep neural networks (DNN) are at the forefront of ongoing research in object detection and classification [8]–[11].

In the fields of image processing and computer vision, the exploration of alternative approaches to traditional square-based systems is not only a technological pursuit but often an attempt to mirror the underlying principles of nature itself (Fig. 2). ANNs, for instance, are biologically inspired in structure and function [12]–[14]. Yet, conventional square

image data formats used in recording and output devices also present limitations [15]–[18]. The human visual system offers an alternative that manifests itself with the sensory cells of the human eye and its retina in the shape of hexagonal arrangements [19], [20] (Fig. 1). Therefore, the use of hexagonal structures emerges as an evolutionary-inspired approach [21] within image processing. The hexagonal lattice format shows several advantages over its square equivalent (Fig. 1), including its homogeneity, uniqueness of neighborhood, and increased radial symmetry, leading to a 13.4 % higher sampling density [20], [22], [23], which can result in reduced computation times, less quantization errors, and an increased efficiency in programming [16], [24]. However, a comprehensive design, implementation, and evaluation of hexagonal imagery has yet to be realized [20], [25], [26].

(a) High density

(b) Low density

(c) Square lattice format

(d) Hexagonal lattice format

(e) Original image

(f) Square lattice format based segment

(g) Hexagonal lattice format based segment

Figure 1: (a,b) Segment of the retina of the human eye (reprinted from [19], copyright AAAS). (c,d) Square and hexagonal lattice format in comparison. (f,g) Square and hexagonal imagery with (e) the test image *Mandrill* from the USC-SIPI image database [27]. The hexagonal lattice format features up to three axes, whereas the *z*-axis can be seen as an additional means to addressing pixels.

A. RELATED WORK

In image processing, early explorations of hexagonal images included comparisons of image quality between square and hexagonal lattices using Bresenham's line algorithm [28], [29], noting fewer artifacts with hexagonal representations. *Overington* [30] investigated hexagonal formats in computer vision, discussing their limitations and potentials. Subsequently, hexagonal addressing schemes were developed for the storage and visualization of hexagonal image representations [20], [31], [32]. However, as true hexagonal images are, in essence, non-existent, image transformation techniques were crucial for converting square imagery into a hexagonal representation [33], including image interpolation approaches [31], [34].

[1]AlexanderAlUS, CC BY-SA 3.0, via Wikimedia Commons (WC)
[2]Alexey Kljatov, CC BY 4.0, WC
[3]Gemma Stiles from Sydney, Australia, CC BY 2.0, WC
[4]David Hawgood / Hex. (mainly) at the Giant's Causeway, CC BY-SA 2.0, WC
[5]NASA/JPL-Caltech/Space Science Institute, CC BY 2.0, WC
[6]Emmanuel Boutet, CC BY-SA 3.0, WC
[7]Opo Terser, CC BY 2.0, WC
[8]Xincailiao.com, CC BY-SA 4.0, WC
[9]CTA/M-A. Besel/IAC (G.P. Diaz)/ESO, CC BY 4.0, WC

(a) Graphene[1]

(b) Snowflakes[2]

(c) (Soap) bubbles[3]

(d) Basalt columns[4]

(e) Cloud formations at the north pole of Saturn[5]

(f) Honeycombs[6]

(g) Compound eyes[7]

(h) Honeycomb materials[8]
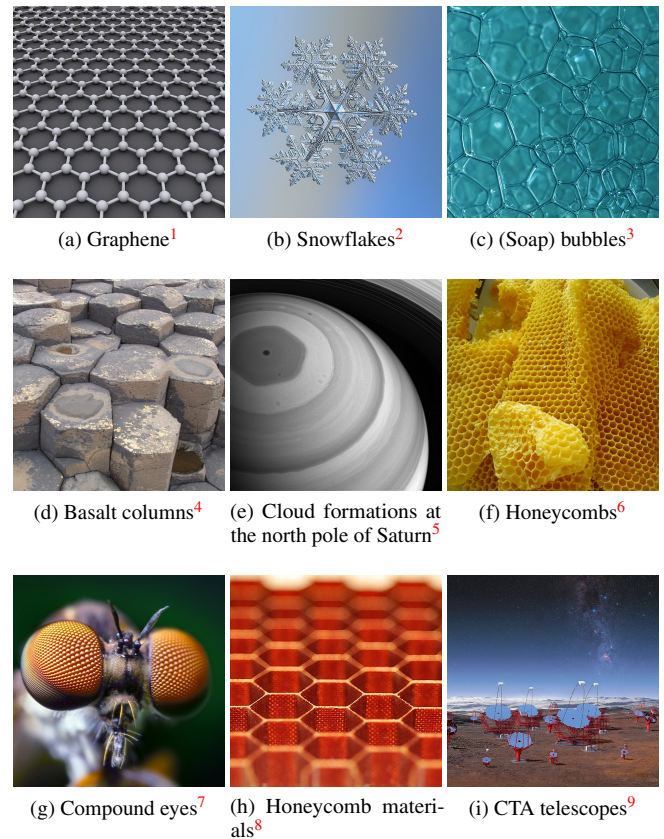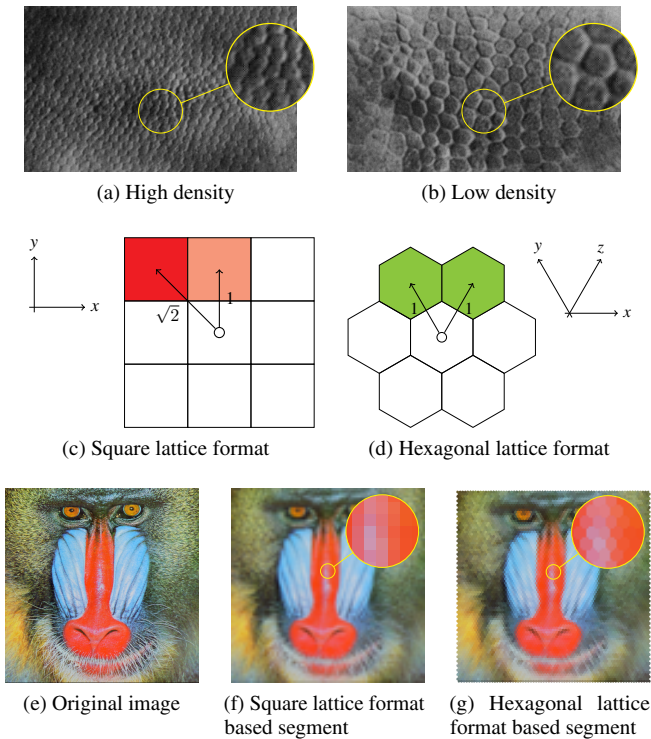
(i) CTA telescopes[9]

Figure 2: Exemplary overview of hexagons and hexagonal lattice formats in nature as well as man-made applications.

In application, hexagonal structures have been developed for tasks such as image segmentation [35], image sampling [36], and atmospheric imaging [37], [38] (Fig. 4), extending to fields of ecology [39], geodesic grid systems [40], sensor-based image processing [36], [41], [42], medical imaging [43]–[45], and image synthesis [46] (Fig. 3). Experimental applications in supply-demand prediction [47] and atmospheric telescope data analysis [48], [49] demonstrate the expanding scope of hexagonal image processing. These aim to advance hexagonal machine learning by addressing the limited integration of hexagonal image processing principles in deep learning (DL), specifically focusing on hexagonal DNNs [50]–[53].

More recent contributions have shown the performance of hexagonal models in domains such as steganography (concealed representation of information), where it resulted in an improved image quality in comparison to its square counterpart [54]. In face recognition, hexagonal approaches to gray level co-occurrence matrices (GLCM), local binary patterns (LBP), and local-holistic graph-based descriptors (LHGPD) have shown improved accuracy compared to their square baseline for face recognition [55]. In image segmentation, hexagonal models show improved segmentation accuracies [56], [57], which "better captures spatial relationships in medical images, leading to improved segmentation performance with potential

IEEE Access·

(a) Image genera-
tion (repr. from [28,
FIG. A1], © Else-
vier)

(b) Image sensors
(repr. from [36, Fig-
ure 4], © Editions
Frontières)

(c) Medical
imaging (repr.
from [45, Figure
2], © IEEE)

(d) Image filtering
(repr. from [20, Fig.
4.11], © Springer)

(e) Edge detection
(repr. from [25, Fig.
11f], © Elsevier)

(f) Image segmen-
tation (repr. from
[35, Figure 8], ©
IEEE)

(g) Geodesic grid
systems (repr. from
[40, Figure 12a],
© Taylor & Fran-
cis)

(h) Astronomical
image processing
(repr. from [37,
Figure 1a], ©
CUP)

(i) Supply-demand
prediction (repr.
from [47, Fig. 8b],
© IEEE)

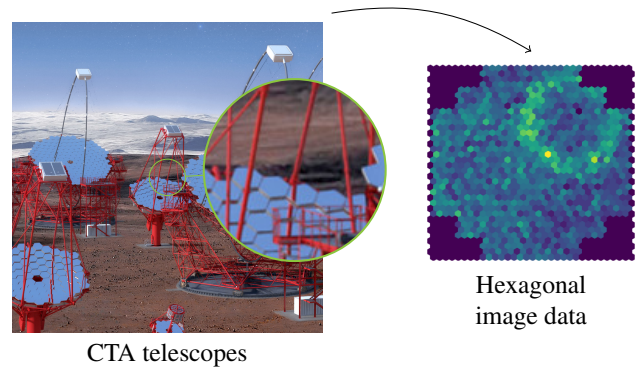Figure 3: Related work on hexagonal image processing.

CTA telescopes

Hexagonal
image data

Figure 4: Cherenkov Telescope Array (CTA) illustration
showing different exemplary hexagonal telescope arrays.[10]

clinical applications" [57]. In remote sensing for grid mod-
eling, sampling, quantization, and storage [58], hexagonal
representations show improved image quality with reduced
sampling artifacts. With more fundamental research being
conducted, approaches to image representation [59], [60],
image transformation and sampling [61], [62], and image
filtering [63] are being further developed, demonstrating the
advantages of hexagonal imagery in terms of image quality
and feature detection.

B. CONTRIBUTION OF THIS WORK

Despite the recognized potential of hexagonal approaches in
image processing and computer vision [20], [52], the first step
towards the development of hexagonal image data sets has
yet to be taken. This contribution aims to bridge this gap by
proposing a baseline for the generation of hexagonal images.
For this purpose, the realized *Hexnet Dataset* and its image
generators are introduced, which feature hexagonal images
for geometric primitives, astronomical image processing,
and hexagonally transformed conventional image data sets.
Furthermore, the methodologies utilized in the data set's

generation are detailed, enabling their reproducibility and
modification with varying parameters to foster their further
development.

**The novel contributions and findings of this work are
summarized as follows:**

1) The publication of hexagonal imagery in the form of
   hexagonal image data sets and their generators, for
   which common image file formats are utilized. These
   include application areas and use cases for..

   a) ..geometric primitives, such as lines, curves, el-
      lipses, grids, and their combinations.
   b) ..astronomical image processing with different
      visualized atmospheric event classes.
   c) ..conventional square image data sets, including CI-
      FAR, CINIC, COIL, MNIST, and Tiny ImageNet.

2) Preliminary investigations on the possible benefits of
   hexagonal imagery. It could be observed that hexagonal
   imagery can results in..

   a) ..improved storage requirements,
   b) while showing an improved image quality.
   c) ..improved model performance for tasks in appli-
      cation, such as image classification.

3) The deployed generators' functionality is detailed to
   enable the further development of hexagonal image
   data and their evaluation. To facilitate the generation
   of hexagonal imagery, all data sets, generators, and
   documentations are published.

For review purposes, the *Hexnet Dataset* is available via
the Chemnitz University of Technology[11]. It was generated
using The *Hexagonal Image Processing Framework Hexnet*
(*Hexnet Framework*). Our implementations can be found via
Hexnet's project page and repository[12].

---

[10]CTA/M-A. Besel/IAC (G.P. Diaz)/ESO, © ① 4.0, via Wikimedia Com-
mons

[11]The *Hexnet Dataset*, https://tuc.cloud/index.php/s/mdsQY3RYgb7YCax
[12]The *Hexagonal Image Processing Framework Hexnet*, https://github.
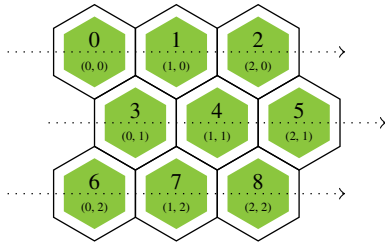com/TSchlosser13/Hexnet

Figure 5: Storage of hexagonal image data.

C. SECTION OVERVIEW

The following sections give an overview of the fundamentals and implementation of hexagonal image generation principles (section II) and the *Hexnet Dataset* and its image generators (section III). Subsequently, additional documentations on hexagonal image generation and visualization are provided (section A), including the *Hexnet Dataset*'s intended uses (section B) and its hosting, licensing, maintenance, and ethics considerations. Finally, further examples on image generation are illustrated.

II. FUNDAMENTALS AND IMPLEMENTATION

To facilitate the generation, processing, storage, and visualization of hexagonal images using the hexagonal lattice format [64], it is essential to implement its underlying structure and addressing scheme [59], [65], [66]. To enable a simplified storage of hexagonal images, Fig. 5 shows our proposed implementation, where a one-dimensional linewise architecture – which is common for square images – is employed. With this implementation, hexagonal images can be viewed using Hexnet's graphical user interface for visualization purposes.

The subsequent sections introduce our image generation methodologies: (i) *geometric primitives* for the evaluation of hexagonal structures and (ii) *astronomical image processing* for atmospheric event detection and classification.

A. GEOMETRIC PRIMITIVES

The following sections give an overview of the relevancy of generating hexagonal images and our related methodology for the generation of geometric primitives with the hexagonal lattice format.

1) Relevancy

To enable the assessment of hexagonal images, it is imperative to establish a baseline data corpus that enables the evaluation of hexagonal approaches to image operations. Given the current lack of such hexagonal imagery, the *TESTIMAGES archive* [67], [68] can be understood as a viable square alternative. For the purpose of hexagonal image generation, a set of basic geometric shapes and contours is defined, for which image transformation techniques, particularly those employing interpolation methods, facilitate the sampling process for the approximation of hexagonal imagery. Within Fig. 6, our underlying motivation is illustrated for the sampling of a test object given both lattice formats. In the shown example, both

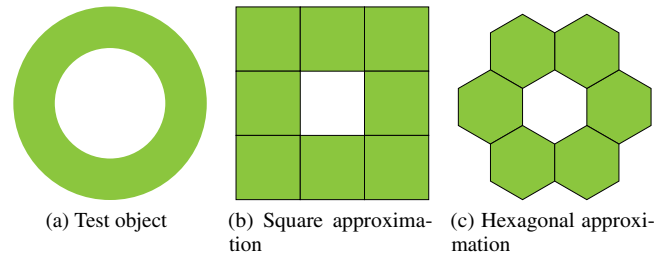| (a) Test object	(b) Square approximation	(c) Hexagonal approximation

Figure 6: Exemplary sampling of a geometric primitive, showcasing a favorable hexagonal approximation with reduced pixel count and possibly less approximation errors.

approximations highlight the potentially resulting advantages. For instance, in a best-case scenario, a resulting hexagonal image may reduce the number of required pixels while reducing approximation errors.

2) Methodology

Our hexagonal image generation methodology involves representing basic geometric primitives through mathematical functions, which are then sampled using the square and the hexagonal lattice format (Fig. 7). This process is governed by different parameters:

- **Mathematical functions.** Mathematical functions and their symbols that are utilized to visualize different classes of geometric primitives such as lines, curves, and ellipses.
- **Figure sizes, step sizes, and linewidths.** Image sizes, step sizes for the sampling process, and sampling linewidths.
- **Rotations, function factors, and translations.** Function rotations, modifications such as function skewness, and translations in order to enable further function modifications.

Illustrated in Fig. 8, the sequence of processing steps for hexagonal image generation is delineated. Within Hexnet, the handling and manipulation of mathematical functions and symbols is realized using the SymPy computer algebra (symbolic computation) library [69]. The sampling process includes the parameters step size, linewidth, and a minimum search radius of one pixel within the target image. For all following investigations, a nearest-neighbor interpolation kernel is utilized, for which binary images are generated. However, it is also noted that further interpolation kernels [31], [34] can be used to generate grayscale or even colored images.

B. ASTRONOMICAL IMAGE PROCESSING

The following sections give an overview of the relevancy of generating hexagonal images and our related methodology for the generation of astronomical images with the hexagonal lattice format.
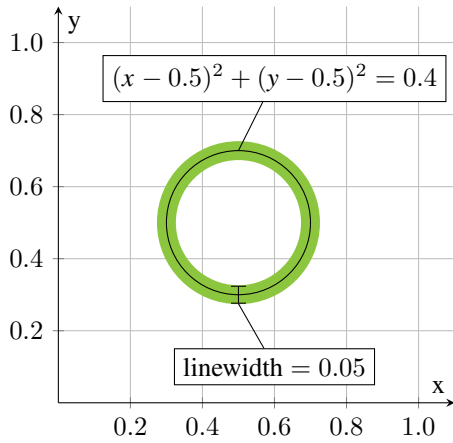
IEEE Access

Figure 7: An exemplary linewidth of 0.05 determines the width of the shown circular geometric primitive. In the background, the square lattice format with its image dimension is displayed.
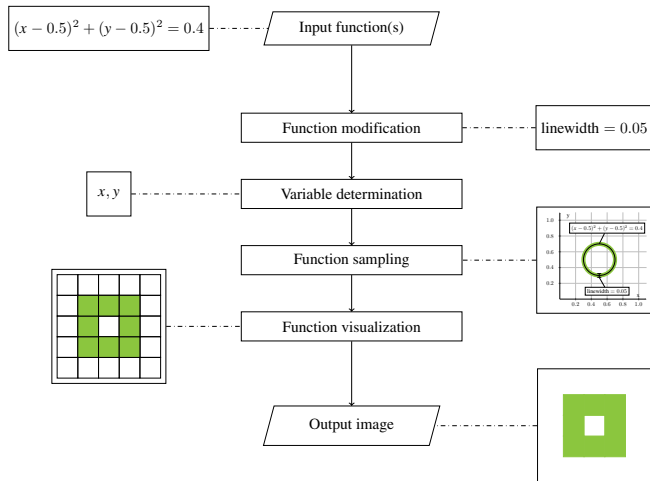

Figure 8: Processing steps for geometric primitive generation.

1) Relevancy

In the field of astronomical image processing, the focus of analyzing atmospheric telescope data centers around the detection and classification of atmospheric events [48], [49]. Projects such as the Cherenkov Telescope Array (CTA)[13], which are often noted for their scientific potential in developing next-generation gamma-ray detectors, are of central importance in this domain. Previous research emphasized the importance of distinguishing gamma rays from cosmic-ray (CR) particles due to their direct dependence on the sensitivity of the instrument [37]. In Fig. 4, such hexagonal telescope arrays are illustrated. For effective gamma-ray-based event observation, notably in the analysis of so-called shower areas, it is essential that the resulting image data is amenable [37], [38], [49].
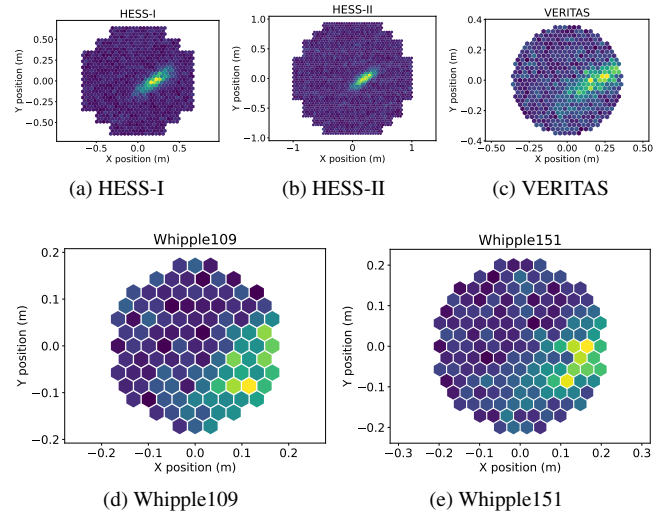

Figure 9: Exemplary telescope array overview of the selected hexagonal telescope arrays HESS-I (a), HESS-II (b), VERITAS (c), Whipple109 (d), and Whipple151 (e).

2) Methodology

For image generation, we employ the Prototype CTA Pipeline Framework (ctapipe)[14], which is designed for prototyping low-level data processing algorithms [70]. It facilitates the simulation and generation of hexagonal images based on various CTA telescope array layouts, including square and hexagonal arrangements, each reflecting the characteristics of their real-world counterparts. With their arrangements, different telescope arrays feature varying sizes of sensory elements. Specifically, five hexagonal telescope arrays – *HESS-I*, *HESS-II*, *VERITAS*, *Whipple109*, and *Whipple151* – were chosen for their uniformly-sized sensory elements, enabling their representation within a single hexagonal image (Fig. 9).

For simulation, different generators for photoelectron distributions across telescope arrays can be applied. This includes imaging functionality for Gaussian, skewed Gaussian, and ring Gaussian distributions of shower areas (Fig. 11) [71], for which image generation parameters such as the shower area's centroid, width, length, and orientation are determined. In terms of distributions, Gaussian and skewed Gaussian distributions can be used to generate general event images, while ring Gaussian distributions models simplified muon rings [72], [73]. Additional parameters include the resulting image's intensity (total number of photoelectrons) and the night sky background's (NSB) [74] distribution *nsb_level_pe*.

III. THE HEXNET DATASET: IMAGE GENERATION

Based on the previously discussed fundamentals of hexagonal image generation, different data subsets for geometric primitives and astronomical images are generated. For this purpose,

---

[13]Cherenkov Telescope Array, https://www.cta-observatory.org/

[14]ctapipe documentation, https://ctapipe.readthedocs.io/en/latest/

Table 1: Shown are the proposed functions and parameters for primitive generation, separated into figure sizes, step sizes, linewidths, rotation degrees, function factors, and translations. In parentheses, the number of resulting values per parameter are denoted.

Parameter	Functions / parameter range
Functions f1–f6 (6)	f1: $\{x\}$
	f2: $\{\sqrt{x}\}$
	f3: $\{1/4 \cdot (2 - \sqrt{-3 + 16 \cdot x - 16 \cdot x^2}),$ $1/4 \cdot (2 + \sqrt{-3 + 16 \cdot x - 16 \cdot x^2})\}$
	f4: $\{-x + 1 - 0.4, -x + 1 - 0.3, \ldots, -x + 1 + 0.4\}$
	f5: $\{-\sqrt{x} + 1 - 0.4, -\sqrt{x} + 1 - 0.3, \ldots, -\sqrt{x} + 1 + 0.4\}$
	f6: Combinations of f1–f3
Figure sizes (4)	$\{60 \times 60, 80 \times 80, 100 \times 100, 200 \times 200\}$
Step sizes (5)	$\{0.001, 0.002, \ldots, 0.005\}$
Linewidths (5)	$\{0.01, 0.02, \ldots, 0.05\}$
Rotation degrees (6)	$\{0, 60, \ldots, 300\}$
Function factors (2)	$\{1.0, 1.2\}$
Translations (9)	$\{(-0.25, -0.25), (-0.25, 0.0), \ldots, (0.25, 0.25)\}$

different classes and their corresponding parameterizations are proposed. In addition, hexagonally transformed versions of commonly evaluated square image data sets are provided.
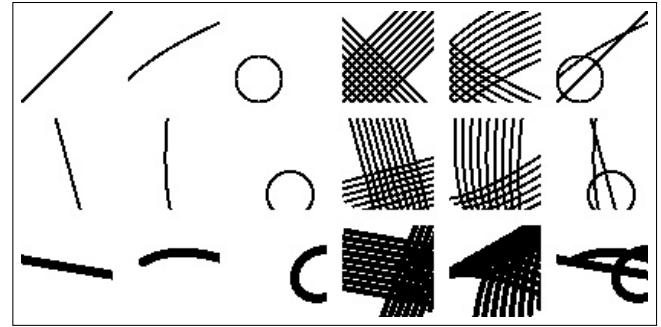
A. GEOMETRIC PRIMITIVES

Table 1 presents an overview of the parameter ranges designated for the image generation of geometric primitives. These parameters encompass different base functions for the classes lines (f1), curves (f2), ellipses (f3), line-based grids (f4), curve-based grids (f5), and different combinations of f1–f3 (f6). Additionally, the related specific mathematical base functions, figure sizes, step sizes, linewidths, rotation degrees, function modifiers, and translations employed are listed to facilitate the generation and subsequent utilization of images sampled via both formats.

An overview of the generated classes is provided in Fig. 10. For the first three examples of the first row we obtain: x (parameterization – step size: 0.001, linewidth: 0.01, rotation: 0°), $\sqrt{x + 0.25} - 0.25$ (same parameterization), and $\{1/4 \cdot (2 - \sqrt{-3 + 16 \cdot (x + 0.25) - 16 \cdot (x + 0.25)^2}), 1/4 \cdot (2 + \sqrt{-3 + 16 \cdot (x + 0.25) - 16 \cdot (x + 0.25)^2})\}$ (same parameterization). 2 700 images have been generated per image resolution and class, resulting in 64 800 total images.

B. ASTRONOMICAL IMAGE PROCESSING

Table 2 presents an overview of the parameter ranges designated for astronomical image generation using the distribution functions Gaussian, skewed Gaussian, and ring Gaussian, for which images without, single, and multiple shower areas are distinguished. 1 000 images have been generated per telescope array and class, resulting in 35 000 total images. The parameters for each image sample were randomly selected from the specified ranges given a uniform normal distribution. To mitigate the effect of event overlap, minimum event distances of 0.125 (Gaussian and skewed Gaussian) and 0.3 (ring Gaussian) are maintained. For an overview see Fig. 11. Finally,

(a) Exemplary samples visualized based on the square lattice format
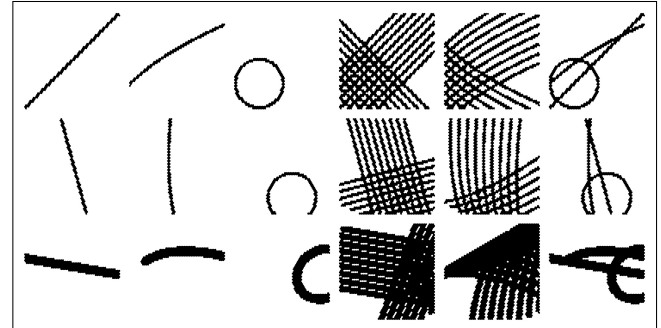
(b) Exemplary samples visualized based on the hexagonal lattice format

Figure 10: Generated exemplary geometric primitive classes with randomized samples for the classes lines, curves, ellipses, line-based grids, curve-based grids, and lines, curves, and ellipses (left to right) for an image resolution of 60×60.

Table 2: Proposed parameter ranges for astronomical image generation with different Gaussian distributions for the sampling process. For further information on the underlying concepts, see [71]–[74].

Distribution	Parameter	Parameter range
all	centroid in meters (x, y)	$[-0.8, 0.8]$
Gaussian and skewed Gaussian	width in meters	$[0.05, 0.075)$
	length in meters	$[0.1, 0.15)$
	orientation ψ in radians	$[0, 2 \cdot \pi)$
skewed Gaussian	skewness γ_1	$[0.1, 0.9]$
ring Gaussian	outer radius in meters	$[0.1, 0.5)$
	inner radius σ in meters	$[0.05, 0.25)$
Image generation		
all	intensity	$[1000, 3000)$
	nsb_level_pe	5

to minimize the effects of image artifacts by the night sky background noise, a common ctapipe-based postprocessing pipeline is employed (Fig. 12).

C. CONVENTIONAL SQUARE IMAGE DATA SETS

The conventional square image data sets CIFAR-10 and CIFAR-100 [75], CINIC-10 [76], COIL-20 [77], COIL-100 [78], MNIST [79], Tiny ImageNet [80], and the USC-SIPI image database [27] have been hexagonally transformed, for which a bilinear interpolation has been deployed. For all data sets, hexagonally transformed and un-/scaled versions are
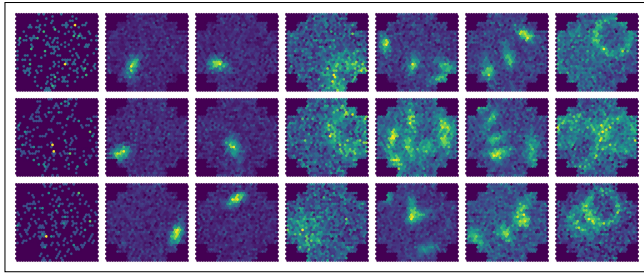
Figure 11: Generated astronomical image classes. From left to right, the shown classes are: no shower areas (1.); single shower area with Gaussian (2.), skewed Gaussian (3.), and ring Gaussian distribution (4.); and multiple shower areas with Gaussian (5.), skewed Gaussian (6.), and ring Gaussian distributions (7.).
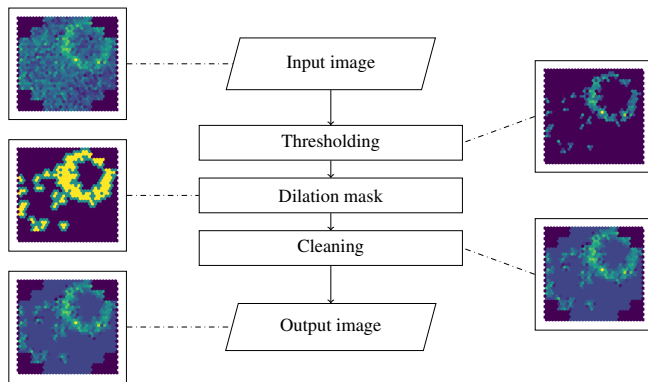

Figure 12: Postprocessing steps for astronomical image processing. To reduce the night sky background's added noise, the input image is cleaned by applying image thresholding and dilation operations.

introduced. For the basic transformed versions, the original image resolutions have been reproduced as closely as possible. These are summarized as follows (*square image resolution* $\rightarrow$ *hexagonal image resolution*): CIFAR-10, CIFAR-100, and CINIC-10 ($32 \times 32 \rightarrow 30 \times 34$), COIL-20 and COIL-100 ($128 \times 128 \rightarrow 119 \times 137$), MNIST ($28 \times 28 \rightarrow 26 \times 30$), Tiny ImageNet ($64 \times 64 \rightarrow 59 \times 69$), and USC-SIPI ($256 \times 256 \rightarrow 238 \times 275$). For visualization purposes, the hexagonal lattice format has been made visible.

IV. CONCLUSION AND OUTLOOK

In this contribution, we tried to expand the traditional boundaries of image data sets, introducing a baseline for hexagonal imagery, The *Hexnet Hexagonal Image Processing Data Set* (short *Hexnet Dataset*). Drawing inspiration from the hexagonal structures in human vision and nature, this data set is meant as a novel approach towards the generation of hexagonal images. Featuring geometric primitives, astronomical image processing, and hexagonally transformed conventional image data sets, the *Hexnet Dataset* can be utilized in exploring the possible advantages of hexagonal structures, for which
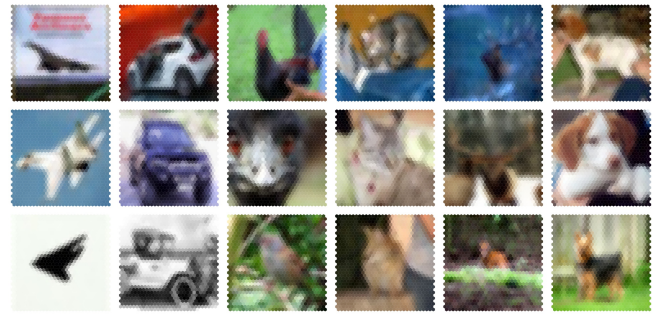
(a) CIFAR-10
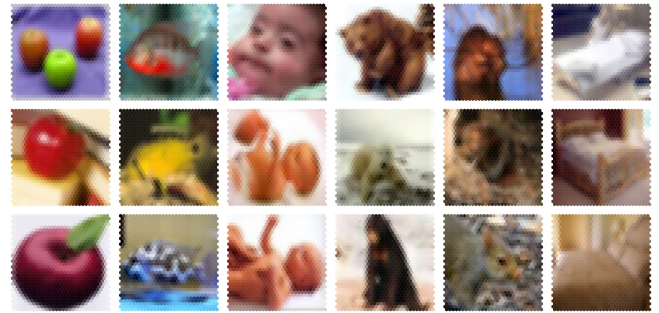
(b) CIFAR-100

Figure 13: Hexagonally transformed conventional square image data sets CIFAR-10 and CIFAR-100 [75].
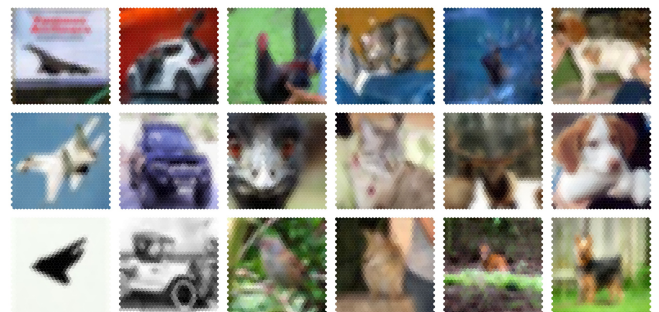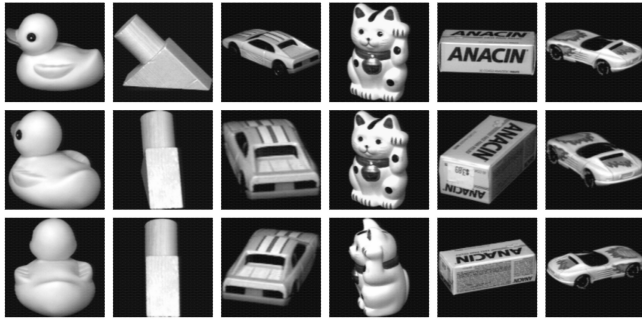

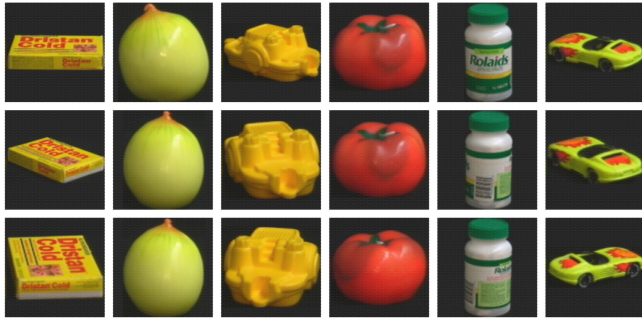Figure 14: Hexagonally transformed conventional square image data set CINIC-10 [76].

common image file formats are used for data storage. However, not only the data set itself but also the related methodologies for generation are provided, enabling its adaptability and further development to promote potentially more efficient yet biologically-aligned computational approaches. For this purpose, our data set is being continuously developed with further parameterizations. Subsequently, further areas of application for the generation of hexagonal images will have to be investigated.

APPENDIX A
DOCUMENTATION

As a baseline, three data subsets are introduced: (i) *geometric primitives* for the evaluation of hexagonal structures, (ii) *astronomical image processing*, in which the descriptions of sensory

(a) COIL-20

(b) COIL-100

Figure 15: Hexagonally transformed conventional square image data sets COIL-20 [77] and COIL-100 [78].
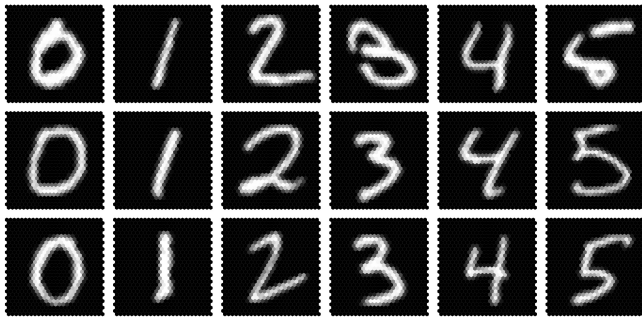

Figure 16: Hexagonally transformed conventional square image data set MNIST [79].

elements of hexagonal telescope arrays have been leveraged for the detection and classification of synthesized atmospheric events, and (iii) *conventional image data sets*, which provides hexagonally transformed versions of commonly evaluated square imagery. They are described in Tables 1 and 2. For data storage, data set directories have been created, whereby our data subsets have been structured given (i) its image resolutions and geometric primitive classes, (ii) its telescope arrays and shower area classes, and (iii) its corresponding conventional image data sets. For formalization purposes, our data set is described using the JSON-LD standard (JavaScript Object Notation for Linked Data) with schema.org syntax.
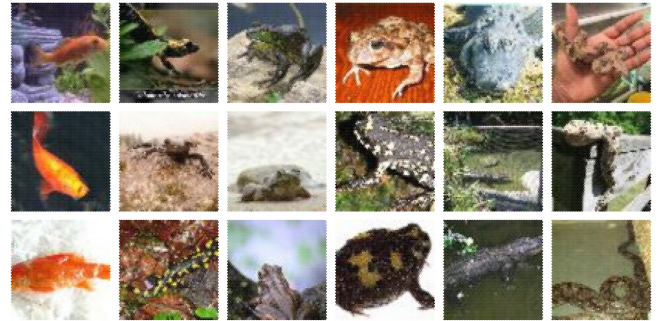

Figure 17: Hexagonally transformed conventional square image data set Tiny ImageNet [80].

For review purposes, the *Hexnet Dataset* is available via the Chemnitz University of Technology[15]. It was generated using The *Hexagonal Image Processing Framework Hexnet* (*Hexnet Framework*). Our implementations can be found via Hexnet's project page and repository[16].

A. IMAGE GENERATION

For hexagonal image generation, a documentation has been created that includes different Jupyter notebooks, introducing hexagonal image processing principles for the generation, processing, storage, and visualization of hexagonal imagery. For more information, see the documentation section of Hexnet on hexagonal image generation (directory `doc/hexagonal_image_generation/`).

For (i), 2 700 images have been generated per image resolution and class, resulting in 64 800 total images. The classes are the functions f1–f6 (Table 1). Our image resolutions are: 60×60, 80×80, 100×100, and 200×200. To ensure reproducibility, all related parameters have been stored within CSV files. Additionally, the number of generated pixels per sample (called *pixels in bound*, short *bip*) is determined. Dependencies for image generation are, e.g., SymPy [69], Matplotlib[17], and joblib[18].

For (ii), 1 000 images have been generated per telescope array and class, resulting in 35 000 total images. The classes are: no shower areas (1.); single shower area with Gaussian (2.), skewed Gaussian (3.), and ring Gaussian distribution (4.); and multiple shower areas with Gaussian (5.), skewed Gaussian (6.), and ring Gaussian distributions (7.) (Table 2). The used telescope arrays are: HESS-I, HESS-II, VERITAS, Whipple109, and Whipple151. To ensure reproducibility, all related parameters have been stored within a CSV file. Dependencies for image generation are, e.g., ctapipe [70] with its dependencies, i.a., Astropy [81], eventio[19], and sim_telarray [82].

---

[15]The *Hexnet Dataset*, https://tuc.cloud/index.php/s/mdsQY3RYgb7YCax
[16]The *Hexagonal Image Processing Framework Hexnet*, https://github.com/TSchlosser13/Hexnet
[17]Matplotlib, https://matplotlib.org/
[18]joblib, https://joblib.readthedocs.io/en/latest/
[19]eventio, https://pypi.org/project/eventio/
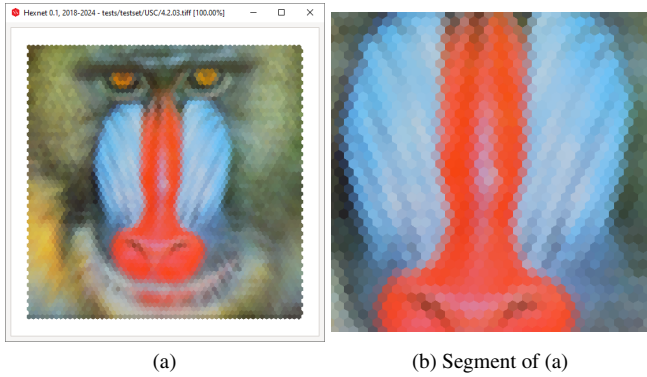
(a)   (b) Segment of (a)

Figure 18: Visualization via Hexnet's GUI.

```
1    ./Hexnet −i tests/testset/USC/4.2.03.tiff −d
```

Listing 1: Visualization via Hexnet's GUI.

For (iii), no changes in terms of image classes, sample counters, or image properties were made to the original image data sets.

B. IMAGE VISUALIZATION

For the visualization of hexagonal images that have been stored using conventional square image file formats, Hexnet's graphical user interface can be used (Fig. 18 and Listing 1). Additionally, a visualization of hexagonal images via Hexnet's image visualization functionality is provided (Listing 2). For more information, please refer to the descriptions of Hexnet's base system (root directory) and its machine learning module (directory _ML/).

APPENDIX B
INTENDED USES

In this section, we would like to provide some observations made during the creation of the *Hexnet Dataset* as well as its possible utilization.

A. PIXEL REQUIREMENTS

A relevant observation stems from the sampling process of the geometric primitives itself. In Table 3, we determined the mean of the total number of pixels that have been sampled per class and function using both lattice formats. With the assessed functions f1–f6, it is observed that, at average, less pixels were

```
1    python _ML/Hexnet.py \
2         −−model \
3         −−dataset $dataset_directory \
4         −−visualize−dataset \
5         [−−visualize−hexagonal \]
6         [−−visualize−colormap viridis \]
7         −−output−dir $output_directory
```

Listing 2: Visualization as raster / vector graphic.

required to sample these functions with the hexagonal lattice format. This is reflected in the resulting reduction of needed pixels, ranging from 0.53 % for ellipses to 4.10 % for lines. Finally, for all generated samples, a mean reduction of needed pixels of 2.47 % is obtained. We intend to discuss these observations in more detail in one of our follow-up contributions. This will include considerations regarding image resolution, image class, and model configuration, training, and testing. Subsequently, further interpolation kernels [31], [34] can be used to generate grayscale or colored images as illustrated in section E.

B. CLASSIFICATION CAPABILITIES

Within the context of object detection and classification, the proposed data subsets can be evaluated to assess the capabilities of learning-based approaches such as hexagonal deep neural networks [50]–[52]. As it is out of this contribution's scope to provide an introduction to hexagonal deep neural networks and their evaluation, we instead would like to present a simplified example. Here, square and hexagonal residual neural networks (S-ResNet and H-ResNet) [83], [84], namely ResNet v1 with three residual stacks with each three residual blocks, have been assessed, for which square convolution kernels of size 3×3 have been replaced by hexagonal ones of size 7 [52]. Consequently, different classification results in incorrectly classified step sizes (a) and linewidths (b) have been observed. Our preliminary results show that distinct classification results for square and hexagonal images can be obtained, whereas different synthesis parameters may result in different advantages or disadvantages for hexagonal images (Fig. 19).

C. IMAGE QUALITY

For the image quality of the square (T_q) and the hexagonal image transformation (T_h), hereinafter also referred to as transformation efficiency T, the transformation efficiency differences $\Delta T = T_h - T_q$ are shown in Fig. 20 using the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) metrics [52], [53], [61], [85]. These results were obtained with the USC-SIPI image database [27]. Each hexagonal pixel was interpolated with its circumradius R, whereas the square lattice format based image transformation was based on the resulting hexagonal arrays' resolution as the minimum square image target resolution. Following [86], test images that are most commonly characterized as natural images, e.g., images (c–f), show evidently increased transformation efficiencies. However, a counterexample can be observed for the test images number 37 (f) and 38 (g), which can a priori not be reliably approximated based on the hexagonal lattice format due to the large number of occurring vertical and horizontal structures. Furthermore, near the circumradius 1, a decline in T_h can be observed. This observation is explained in particular by the resulting square pixel side length, which reaches the image resolution of the original image, therefore resulting in only minor square interpolation artifacts.

Table 3: Mean total pixels sampled per class and function within (i). Shown are the mean total pixels sampled with the square and the hexagonal lattice format. For both lattice formats, the same functions and image generation parameterization has been used. When comparing square and hexagonal results, we can observe a reduction of needed pixels in favor of the hexagonal lattice format, showing a mean reduction in pixels of 2.47 % (97.53 %).

Functions f1–f6 (6)	Mean total pixels sampled [square]	Mean total pixels sampled [hexagonal]	Percentage of needed pixels [hexagonal over square]
Lines (f1)	143.75	**137.85**	**95.90**
Curves (f2)	139.00	**134.60**	**96.84**
Ellipses (f3)	161.35	**160.50**	**99.47**
Line-based grids (f4)	1717.10	**1675.90**	**97.60**
Curve-based grids (f5)	1692.25	**1656.25**	**97.87**
Combinations of f1–f3 (f6)	410.40	**400.15**	**97.50**
		Mean	**97.53**

(a) Incorrectly classified step sizes
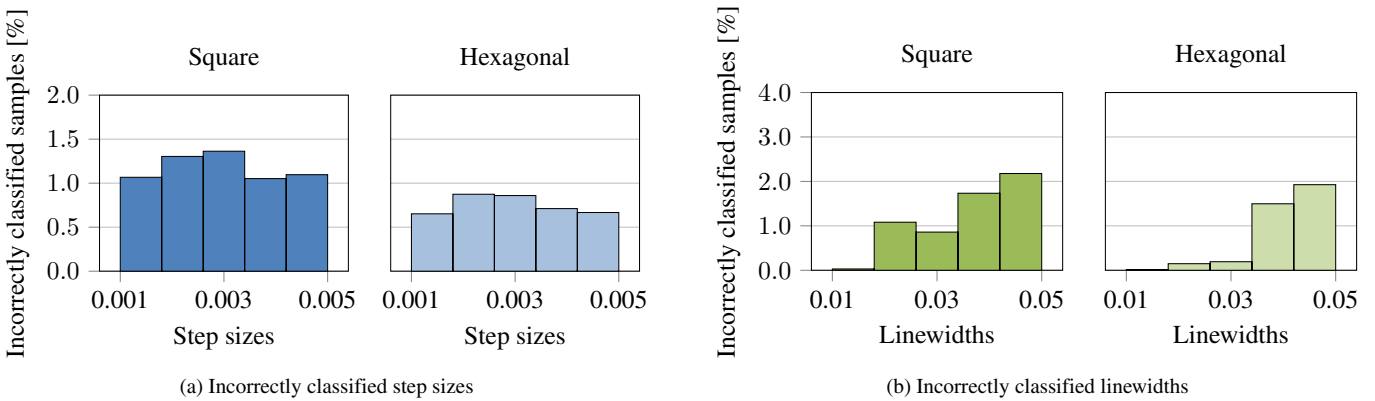

(b) Incorrectly classified linewidths

Figure 19: Incorrectly classified samples per synthesis parameter within (i) for an image resolution of 60×60. Here, square and hexagonal residual neural networks (S-ResNet and H-ResNet) [83], [84], namely ResNet v1 with three residual stacks with each three residual blocks, have been assessed, for which square convolution kernels of size 3×3 have been replaced by hexagonal ones of size 7 [52]. Both models have been trained for 2 epochs. Consequently, different classification results in incorrectly classified step sizes (a) and linewidths (b) have been observed. Overall, the benefits of the hexagonal image data and the hexagonal model are evident.

Following the evaluation of the USC-SIPI image database, the square lattice format based image data sets CIFAR-10 [75], CIFAR-100 [75], CINIC-10 [76], COIL-20 [77], COIL-100 [78], MNIST [79], and Tiny ImageNet [80] were further investigated. For this purpose, Table 4 shows their results overview with the obtained PSNR- (top) and SSIM-based transformation efficiencies (bottom, respectively) in terms of their minimum, maximum, mean, and median scores. Over all results, it is evident that the hexagonal image transformation results in an overall increased transformation efficiency, whereas only USC-SIPI and Tiny ImageNet show a greater minimum than maximum in transformation efficiency. However, as highlighted before for USC-SIPI, these results are also influenced by the occurrence of single images with prevalent square structures.

D. FURTHER APPLICATION AREAS

Further application areas for (i) include tasks such as image segmentation [35] and sampling [36]. Within (ii), our data can be used for atmospheric event detection and classification [37], [38], [48], [49], for which approaches such as hexagonal deep neural networks can be employed [48]–[52]. Application areas that have yet to be investigated regarding the generation of hexagonal imagery include ecology [39], geodesic grid systems [40], sensor-based image processing [36], [41], [42], medical imaging [43]–[45], and image synthesis [46].

APPENDIX C
DISCUSSION

Hexagonal image processing offers advantages in terms of sampling efficiency, neighborhood relationships, and image quality. However, its widespread adoption is contingent upon overcoming hardware limitations and further refining software tools, such as the *Hexnet Framework*, to fully leverage the

(a) PSNR

(b) SSIM

(c) 1   (d) 2   (e) 3   (f) 4   (g) 5   (h) 6

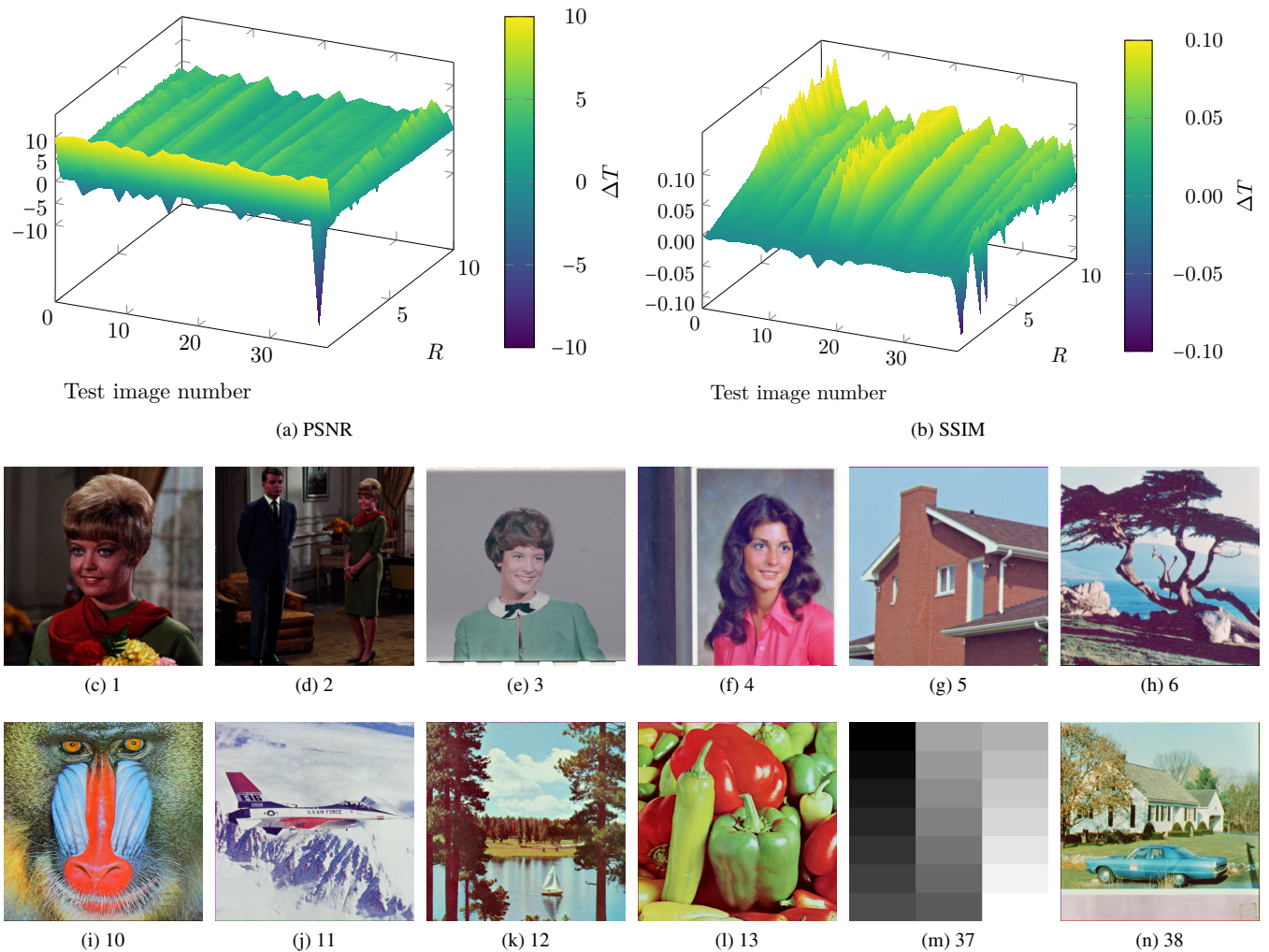(i) 10   (j) 11   (k) 12   (l) 13   (m) 37   (n) 38

Figure 20: Square and hexagonal bilinear interpolation based image transformation comparison with the USC-SIPI image database [27] utilizing the peak signal-to-noise ratio (PSNR) and the structural similarity (SSIM) metrics in subpixel resolution. Each hexagonal pixel was interpolated with its circumradius R of 0.1 to 5.0 in steps of 0.1, whereas the square lattice format based transformation was based on the resulting hexagonal arrays' resolution as the minimum square image target resolution.

benefits of hexagonal imagery in real-world applications. In this section, we address further considerations regarding the properties of hexagonal imagery and its benefits.

A. IMAGE STORAGE AND VISUALIZATION

For the storage of hexagonal imagery, common square image file formats, such as Portable Network Graphics (PNG), are used, which employ lossless compression algorithms such as run-length encoding. These compression methods work independently of the underlying lattice format, as pixel offsets are ignored. Lossy compression algorithms are deliberately avoided, as they could exacerbate compression artifacts due to the distinct nature of hexagonal images compared to square ones.

Common image viewers are unable to render graphics with hexagonal sampling, as only specialized viewers, such as Hexnet's graphical user interface (GUI), are capable of displaying hexagonally encoded images in their native form. For visualization, Hexnet's GUI leverages general-purpose computing on graphics processing units (GPGPU) to optimize the shader-based handling of these images. Users can choose to visualize hexagonal images either in their native hexagonal form via Hexnet or as square images through conventional raster graphic viewers. With Hexnet, hexagonal images can also be rendered as vector graphics, with all the necessary information embedded in the image file to facilitate their display through standard image viewers and processing tools. For raster graphics, additional metadata files accompany the *Hexnet Dataset*, providing further details on the hexagonal specifics of the image generation process to ensure correct decoding.

It is noted that although the raw data are accurately encoded, typical image viewers may render hexagonal images as square due to the inherent interpolation that occurs on standard

Table 4: Square and hexagonal bilinear interpolation based image transformation efficiency comparison utilizing the peak signal-to-noise ratio (PSNR, top) and the structural similarity (SSIM, bottom, respectively) metrics in subpixel resolution for different conventional square lattice format based image data sets. Shown are the resulting ΔT values. Each hexagonal pixel was interpolated with its circumradius R and the radii of 0.1 to 5.0 in steps of 0.1, whereas the square lattice format based transformation was based on the resulting hexagonal arrays' resolution as the minimum square image target resolution.

Data set	Minimum	Maximum	Mean	Median
CIFAR-10 [75]	-4.633 -0.111	12.140 0.207	2.036 0.025	1.880 0.020
CIFAR-100 [75]	-13.764 -0.312	13.433 0.461	2.122 0.022	1.972 0.017
CINIC-10 [76]	-5.476 -0.103	12.476 0.364	2.134 0.026	1.999 0.020
COIL-20 [77]	-10.201 -0.008	11.622 0.115	4.296 0.020	4.451 0.017
COIL-100 [78]	-4.331 -0.008	13.603 0.130	4.836 0.022	5.174 0.019
MNIST [79]	-9.851 -0.173	13.768 0.286	2.076 0.037	1.768 0.031
Tiny ImageNet [80]	-14.461 -0.141	13.244 0.247	2.219 0.028	1.990 0.027
USC-SIPI [27]	-23.961 -0.096	11.596 0.124	3.109 0.026	3.071 0.023

monitors, which is also true for square images when the display resolution does not perfectly match the image resolution. However, this discrepancy is not significant, as it only affects the viewing experience and not the underlying image data.

B. IMAGE QUALITY AND HEXAGONAL SAMPLING EFFICIENCY

The hexagonal grid offers several advantages in image quality and sampling efficiency compared to the square grid. Hexagonal grids exhibit higher radial symmetry and homogeneity, leading to a 13.4 % increase in sampling density, which allows hexagonal grids to store more information with fewer sampling points. Furthermore, the increased symmetry of hexagonal grids minimizes image artifacts, particularly when aligning along multiple orientations.

In nature, sharp lines and edges, or square structures, are rare, with circular, ellipsoid, and curved forms being more prevalent, contrasting with the sharp features often found in rectangular-arranged structures and objects (e.g., industrial manufacturing processes). Hexagonal grids offer increased radial symmetry compared to square ones, which have only 2 axes and a perfect radial symmetry at 90 degrees, allowing image artifacts to be minimized when aligning along 4 orientations. In contrast, the hexagonal grid features 3 axes and a perfect radial symmetry at 60 degrees, enabling alignment along 6 orientations. This advantage suggests that, even when a perfect line at 90 degrees is to be imaged, hexagonal representations offer benefits when considering all possible

orientations. This may also explain the observations made in Table 3, showing that fewer sampling points are required for lines in hexagonal grids.

However, certain scenarios, such as imaging a square object (e.g., a checkerboard surface) from above, may be better suited to square sampling. Yet, when the perspective changes – such as viewing the checkerboard from an angle where the vanishing lines align closer to the 60 degrees symmetry axes – hexagonal sampling could offer advantages even for square objects, depending on their orientation and perspective. This negative phenomenon is reflected in Fig. 20 (m, image 37), demonstrating reduced transformation efficiency (ΔT) for straight lines, particularly rectangles.

C. NEIGHBORHOOD RELATIONSHIPS AND SAMPLING KERNELS

The neighborhood relationships within the square lattice format can result in ambiguity, as two types of neighborhood can be considered. These are denoted as direct and indirect or 4- and 8-neighborhood relationship. This ambiguity increased the amount of neighborhood cases to be considered when, for instance, border or line following algorithms are to be utilized, such as Bresenham's line algorithm [28], [29]. With the hexagonal lattice format, a uniqueness of neighborhood is inherent, which means that no differentiation between direct and indirect neighborhood relationships has to be made, as only direct neighboring pixels exist. Beyond the direct neighborhood, different pixel distances result for both the square and the hexagonal lattice format that have to be considered.

When considering image quality, previous work has shown that hexagonal imagery can result in improved image quality, which is, for instance, measured by the transformation efficiency of square and hexagonal images [52], [53], [61]. For example, *Schlosser et al.* [52], [53] and *Fadaei and Rashno* [61] show that improved image quality can be quantified even when subpixel image resolutions are being considered. For this purpose, an investigation into image similarity is conducted in Fig. 20 and Table 4. With an increasing circumradius R, a stronger downscaling is realized, highlighting the hexagonal benefit in terms of image quality in comparison to the original imagery. This observation may serve as a starting point for further evaluations on sampling methods and their parameterization (e.g., neighborhood definitions and sampling distances).

D. HARDWARE AND SOFTWARE LIMITATIONS

Despite the theoretical and practical advantages of hexagonal grids, the transition from square to hexagonal imagery is hindered by the lack of hexagonal input and output hardware, such as cameras and monitors [20], [25], [26]. While efforts have been made in research and sensor-based image processing [36], [41], [42], large-scale development of hexagonal devices remains limited. Notable exceptions, such as the Cherenkov Telescope Array, could catalyze the development of such hardware. On the software side, the *Hexnet Framework* represents a significant step forward in processing hexagonal image data.

It provides a versatile framework for converting square images to hexagonal formats and simplifies handling hexagonal grids [52], [53]. Moreover, it provides further functionality on specific image processing tasks from general image processing and computer graphics, computer vision, as well as machine learning and deep learning, which will be introduced as part of our future work. As hexagonal sensors remain unavailable, techniques such as downsampling square sensor data to hexagonal resolutions [59] offer a feasible workaround for approximating hexagonal sensor performance, especially when input resolution exceeds processing capabilities.

E. DATA GENERATION AND THE HEXNET DATASET

The generation of hexagonal images with the *Hexnet Dataset* and its generators involves several mathematical functions for producing geometric primitives and astronomical data. The data set includes a variety of image generation kernels, allowing for the creation of lines, curves, grids, and more complex shapes, as well as conventional square image data sets such as CIFAR and Tiny ImageNet. Preliminary results suggest that hexagonal and square approaches exhibit no significant performance differences in terms of image generation [52]. However, the geometric properties of both grids – such as the number of white pixels or line thickness – directly influence computational complexity, with more complex functions requiring longer generation times. Although comprehensive evaluations of image generation kernels are beyond the scope of this work, the provided documentation and illustrations (e.g., Listings 3 to 6 and Fig. 21 to 40) demonstrate how different parameters affect image generation.

F. COMPARATIVE FRAMEWORKS AND THE FUTURE OF HEXAGONAL IMAGE PROCESSING

To our best knowledge, no previous or current research provides implementations that would enable the generation or hexagonal imagery. Instead, their main focus often lies with the underlying principles of hexagonal image processing itself [47], [54]–[56], [58], [59].

However, in terms of general hexagonal image processing frameworks, one alternative exists, the hexagonal image processing framework HIP (HIP framework) [20]. It provides functionality for the storage, processing, and visualization of hexagonal imagery, including the square-to-hexagonal and hexagonal-to-square image transformation, which is based on image interpolation, as well as functionality to determine pixel neighborhoods and their distances. However, since its conception, the HIP framework has been extended and optimized [18], with previous work highlighting its limitations in terms of image quality [18] and performance [18], [52] in comparison to the *Hexnet Framework* [52]. The *Hexnet Framework* provides this functionality as core functionality while providing further functionality on specific image processing tasks from general image processing and computer graphics, computer vision, as well as machine learning and deep learning. However, it is out of its scope to introduce the *Hexnet Framework* and its evaluation, which will be a part of our future work.

APPENDIX D
HOSTING, LICENSING, MAINTENANCE, AND ETHICS

For data set hosting purposes, the file hosting platform of the Chemnitz University of Technology will be used to provide our data set. Additionally, we plan on publishing our data set via the open access repository Figshare[20] to be eligible to be indexed by the Google Dataset Search[21]. We are, however, also open to alternative suggestions regarding data hosting possibilities. The *Hexnet Dataset* will be licensed via ⓒ① 4.0. For its maintenance, a continuous maintenance with further extensions of our data set, including further parameterizations and classes, is intended. Finally, we declare that this submission is, to the best of our knowledge, free from ethical implications, whereby it aligns with common ethical norms.

APPENDIX E
EXAMPLES

Listings 3 to 6 and Fig. 21 to 24 give an overview of our realized functionality for the generation of geometric primitives. Shown are examples for lines, curves, ellipses (circle), and further functions (examples 1 to 4). Fig. 25 to 34 detail the influence of the different image generation parameters, including figure sizes, step sizes, linewidths, rotation degrees, and function factors. Fig. 35 to 40 show additional atmospheric event images generated with varying distributions, intensities, and number of events for astronomical image generation.

ACKNOWLEDGMENT

AUTHOR CONTRIBUTIONS

Tobias Schlosser conducted this contribution's writing process and the related research project's implementation and evaluation with the help of Michael Friedrich, Trixy Meyer, and Danny Kowerko in revising this manuscript. Tobias Schlosser as well as Maximilian Eibl and Danny Kowerko designed and supervised the related research project.

CONFLICTS OF INTEREST

The authors declare that they have no conflict of interest.

References

[1] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015. [Online]. Available: https://www.nature.com/articles/nature14539 (1 citation on 1 page: 1)

[2] J. Schmidhuber, "Deep learning in neural networks: An overview," *Neural Networks*, vol. 61, pp. 85–117, 2015. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0893608014002135 (1 citation on 1 page: 1)

[3] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org (1 citation on 1 page: 1)

[4] T. J. Sejnowski, *The Deep Learning Revolution*. MIT Press, 2019. [Online]. Available: https://mitpress.mit.edu/9780262038034/the-deep-learning-revolution/ (1 citation on 1 page: 1)

[20] Figshare, https://figshare.com/
[21] Google Dataset Search, https://datasetsearch.research.google.com/

[5] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Advances in Neural Information Processing Systems 25*, 2012, pp. 1097–1105. [Online]. Available: http://papers.nips.cc/paper/4824-imagenet-classification-with-deep-convolutional-neural-networ
(1 citation on 1 page: 1)

[6] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich, "Going Deeper with Convolutions," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 1–9. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/html/Szegedy_Going_Deeper_With_2015_CVPR_paper.html
(1 citation on 1 page: 1)

[7] M. M. Bronstein, J. Bruna, Y. LeCun, A. Szlam, and P. Vandergheynst, "Geometric Deep Learning: Going beyond Euclidean data," *IEEE Signal Processing Magazine*, vol. 34, no. 4, pp. 18–42, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7974879
(1 citation on 1 page: 1)

[8] B. Zhao, J. Feng, X. Wu, and S. Yan, "A survey on deep learning-based fine-grained object classification and semantic segmentation," *International Journal of Automation and Computing*, vol. 14, no. 2, pp. 119–135, 2017. [Online]. Available: https://link.springer.com/article/10.1007/s11633-017-1053-3
(1 citation on 1 page: 1)

[9] Z. Q. Zhao, P. Zheng, S. T. Xu, and X. Wu, "Object Detection with Deep Learning: A Review," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 11, pp. 3212–3232, 2019. [Online]. Available: https://ieeexplore.ieee.org/document/8627998
(1 citation on 1 page: 1)

[10] Y. Liu, P. Sun, N. Wergeles, and Y. Shang, "A survey and performance evaluation of deep learning methods for small object detection," *Expert Systems with Applications*, vol. 172, p. 114602, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0957417421000439
(1 citation on 1 page: 1)

[11] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, and B. Lee, "A survey of modern deep learning based object detection models," *Digital Signal Processing*, vol. 126, p. 103514, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1051200422001312
(1 citation on 1 page: 1)

[12] F. Rosenblatt, "The Perceptron—a perceiving and recognizing automaton," *Report 85, Cornell Aeronautical Laboratory*, 1957. [Online]. Available: https://blogs.umass.edu/brain-wars/files/2016/03/rosenblatt-1957.pdf
(1 citation on 1 page: 1)

[13] ——, "The perceptron: A probabilistic model for information storage and organization in the brain," *Psychological Review*, vol. 65, no. 6, pp. 386–408, 1958. [Online]. Available: https://psycnet.apa.org/record/1959-09865-001
(1 citation on 1 page: 1)

[14] T. Manning, R. D. Sleator, and P. Walsh, "Biologically inspired intelligent decision making: A commentary on the use of artificial neural networks in bioinformatics," *Bioengineered*, vol. 5, no. 2, pp. 80–95, 2013. [Online]. Available: https://www.tandfonline.com/doi/full/10.4161/bioe.26997
(1 citation on 1 page: 1)

[15] R. C. Staunton and N. Storey, "A Comparison Between Square and Hexagonal Sampling Methods for Pipeline Image Processing," in *Optics, Illumination, and Image Sensing for Machine Vision IV*, vol. 1194, 1990, pp. 142–151. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/1194/0000/A-Comparison-Between-Square-and-Hexagonal-Sampling-Methods-for-Pipeline/10.1117/12.969847.short
(1 citation on 1 page: 1)

[16] X. He and W. Jia, "Hexagonal structure for intelligent vision," in *International Conference on Information and Communication Technologies (ICICT)*. IEEE, 2005, pp. 52–64. [Online]. Available: https://ieeexplore.ieee.org/document/1598543
(2 citations on 1 page: 1)

[17] X. He, T. Hintz, Q. Wu, H. Wang, and W. Jia, "A new simulation of spiral architecture," in *Proceedings of the 2006 International Conference on Image Processing, Computer Vision, and Pattern Recognition, IPCV'06*, vol. 2, 2006, pp. 570–575. [Online]. Available: https://opus.lib.uts.edu.au/handle/10453/6883
(1 citation on 1 page: 1)

[18] T. Schlosser, R. Manthey, and M. Ritter, "Entwurf und Implementierung von Optimierungs- und Funktionserweiterungen der hexagonalen Bildrasterung in der Videokompressionssoftware x264HMod," in *Studierendensymposium Informatik 2016 der TU Chemnitz*, 2016, pp. 63–74. [Online]. Available: https://nbn-resolving.org/urn:nbn:de:bsz:ch1-qucosa-201104
(4 citations on 2 pages: 1 and 13)

[19] C. A. Curcio, K. R. J. Sloan, O. Packer, A. E. Hendrickson, and R. E. Kalina, "Distribution of Cones in Human and Monkey Retina: Individual Variability and Radial Asymmetry," *Science*, vol. 236, no. 4801, pp. 579–582, 1987. [Online]. Available: https://science.sciencemag.org/content/236/4801/579
(2 citations on 2 pages: 1 and 2)

[20] L. Middleton and J. Sivaswamy, *Hexagonal Image Processing: A Practical Approach*. Springer, 2005. [Online]. Available: https://link.springer.com/book/10.1007/1-84628-203-9
(8 citations on 5 pages: 1, 2, 3, 12, and 13)

[21] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of Physiology*, vol. 195, no. 1, pp. 215–243, 1968. [Online]. Available: https://physoc.onlinelibrary.wiley.com/doi/abs/10.1113/jphysiol.1968.sp008455
(1 citation on 1 page: 1)

[22] D. P. Petersen and D. Middleton, "Sampling and Reconstruction of Wave-Number-Limited Functions in N-Dimensional Euclidean Spaces*," *Information and Control*, vol. 5, no. 4, pp. 279–323, 1962. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0019995862906332
(1 citation on 1 page: 1)

[23] R. M. Mersereau, "The Processing of Hexagonally Sampled Two-Dimensional Signals," *Proceedings of the IEEE*, vol. 67, no. 6, pp. 930–949, 1979. [Online]. Available: https://ieeexplore.ieee.org/document/1455625
(1 citation on 1 page: 1)

[24] M. J. E. Golay, "Hexagonal Parallel Pattern Transformations," *IEEE Transactions on Computers*, vol. C-18, no. 8, pp. 733–740, 1969. [Online]. Available: https://ieeexplore.ieee.org/document/1671349
(1 citation on 1 page: 1)

[25] L. Middleton and J. Sivaswamy, "Edge detection in a hexagonal-image processing framework," *Image and Vision Computing*, vol. 19, no. 14, pp. 1071–1081, 2001. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0262885601000671
(3 citations on 3 pages: 1, 3, and 12)

[26] ——, "Framework for practical hexagonal-image processing," *Journal of Electronic Imaging*, vol. 11, no. 1, pp. 104–114, 2002. [Online]. Available: https://www.spiedigitallibrary.org/journals/journal-of-electronic-imaging/volume-11/issue-1/0000/Framework-for-practical-hexagonal-image-processing/10.1117/1.1426078.short
(2 citations on 2 pages: 1 and 12)

[27] A. G. Weber, "The USC-SIPI Image Data Base: Version 4," University of Southern California, Tech. Rep., 1993. [Online]. Available: http://sipi.usc.edu/reports/pdfs/Scanned/USC-SIPI-244.pdf
(5 citations on 5 pages: 2, 6, 9, 11, and 12)

[28] C. A. Wüthrich and P. Stucki, "An Algorithmic Comparison between Square- and Hexagonal-Based Grids," *CVGIP: Graphical Models and Image Processing*, vol. 53, no. 4, pp. 324–339, 1991. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/104996529190036J
(3 citations on 3 pages: 2, 3, and 12)

[29] J. E. Bresenham, "Algorithm for computer control of a digital plotter," *IBM Systems Journal*, vol. 4, no. 1, pp. 25–30, 1965. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/5388473
(2 citations on 2 pages: 2 and 12)

[30] I. Overington, *Computer Vision: A unified, biologically-inspired approach*. Elsevier, 1992. [Online]. Available: https://dl.acm.org/doi/10.5555/531547
(1 citation on 1 page: 2)

[31] I. Her and C. T. Yuan, "Resampling on a Pseudohexagonal Grid," *CVGIP: Graphical Models and Image Processing*, vol. 56, no. 4, pp. 336–347, 1994. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1049965284710303
(4 citations on 3 pages: 2, 4, and 9)

[32] I. Her, "Geometric Transformations on the Hexagonal Grid," *IEEE Transactions on Image Processing*, vol. 4, no. 9, pp. 1213–1222, 1995. [Online]. Available: https://ieeexplore.ieee.org/document/413166
(1 citation on 1 page: 2)

[33] A. P. Fitz and R. J. Green, "Fingerprint classification using a hexagonal fast fourier transform," *Pattern Recognition*, vol. 29, no. 10, pp. 1587–1597, 1996. [Online]. Available: https://www.sciencedirect.com/science/article/pii/0031320396000180
(1 citation on 1 page: 2)

This article has been accepted for publication in IEEE Access. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2024.3510656

IEEE Access·

T. Schlosser *et al.*: A Hexagonal Image Processing Data Set and Generator

[34] D. Van De Ville, T. Blu, M. Unser, W. Philips, I. Lemahieu, and R. Van de Walle, "Hex-Splines: A Novel Spline Family for Hexagonal Lattices," *IEEE Transactions on Image Processing*, vol. 13, no. 6, pp. 758–772, 2004. [Online]. Available: https://ieeexplore.ieee.org/document/1298833
(3 citations on 3 pages: 2, 4, and 9)

[35] M. Tremblay, S. Dallaire, and D. Poussart, "Low level Segmentation Using CMOS Smart Hexagonal Image Sensor," in *Proceedings of Conference on Computer Architectures for Machine Perception*, 1995, pp. 21–28. [Online]. Available: https://ieeexplore.ieee.org/document/521015
(3 citations on 3 pages: 2, 3, and 10)

[36] R. Hauschild, B. J. Hosticka, S. Muller, and M. Schwarz, "A CMOS Optical Sensor System Performing Image Sampling on a Hexagonal Grid," in *ESSCIRC'96: Proceedings of the 22nd European Solid-State Circuits Conference*, 1996, pp. 304–307. [Online]. Available: https://ieeexplore.ieee.org/document/5468652
(6 citations on 4 pages: 2, 3, 10, and 12)

[37] Q. Feng and T. T. Y. Lin, "The analysis of VERITAS muon images using convolutional neural networks," *Proceedings of the International Astronomical Union*, vol. 12, no. S325, pp. 173–179, 2016. [Online]. Available: https://www.cambridge.org/core/journals/proceedings-of-the-international-astronomical-union/article/analysis-of-veritas-muon-images-using-convolutional-neural-networks/72760373B04B5B76A56120D8E910C312
(5 citations on 4 pages: 2, 3, 5, and 10)

[38] S. Mangano, C. Delgado, M. I. Bernardos, M. Lallena, and J. J. R. Vázquez, "Extracting Gamma-Ray Information from Images with Convolutional Neural Network Methods on Simulated Cherenkov Telescope Array Data," in *IAPR Workshop on Artificial Neural Networks in Pattern Recognition*, 2018, pp. 243–254. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-99978-4_19
(3 citations on 3 pages: 2, 5, and 10)

[39] C. P. D. Birch, S. P. Oom, and J. A. Beecham, "Rectangular and hexagonal grids used for observation, experiment and simulation in ecology," *Ecological Modelling*, vol. 206, no. 3-4, pp. 347–359, 2007. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0304380007001949
(2 citations on 2 pages: 2 and 10)

[40] K. Sahr, D. White, and A. J. Kimerling, "Geodesic Discrete Global Grid Systems," *Cartography and Geographic Information Science*, vol. 30, no. 2, pp. 121–134, 2003. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1559/152304003100011090
(3 citations on 3 pages: 2, 3, and 10)

[41] S. Jung, R. Thewes, T. Scheiter, K. F. Goser, and W. Weber, "A Low-Power and High-Performance CMOS Fingerprint Sensing and Encoding Architecture," *IEEE Journal of Solid-State Circuits*, vol. 34, no. 7, pp. 978–984, 1999. [Online]. Available: https://ieeexplore.ieee.org/document/772413
(3 citations on 3 pages: 2, 10, and 12)

[42] M. Ambrosio, C. Aramo, F. Bracci, P. Facal, R. Fonte, G. Gallo, E. Kemp, G. Matthiae, D. Nicotra, P. Privitera, G. Raia, E. Tusi, and G. Vitali, "The Camera of the Auger Fluorescence Detector," *IEEE Transactions on Nuclear Science*, vol. 48, no. 3, pp. 400–405, 2001. [Online]. Available: https://ieeexplore.ieee.org/document/940088
(3 citations on 3 pages: 2, 10, and 12)

[43] H. Lin, N. J. Wu, F. Geiger, K. Xie, and A. Ignatiev, "A ferroelectric-superconducting photodetector," *Journal of Applied Physics*, vol. 80, no. 12, pp. 7130–7133, 1996. [Online]. Available: https://aip.scitation.org/doi/abs/10.1063/1.363733
(2 citations on 2 pages: 2 and 10)

[44] M. Schwarz, R. Hauschild, B. J. Hosticka, J. Huppertz, T. Kneip, S. Kolnsberg, L. Ewe, and H. K. Trieu, "Single-Chip CMOS Image Sensors for a Retina Implant System," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 46, no. 7, pp. 870–877, 1999. [Online]. Available: https://ieeexplore.ieee.org/document/775382
(2 citations on 2 pages: 2 and 10)

[45] W. Neeser, M. Bocker, P. Buchholz, P. Fischer, P. Holl, J. Kemmer, P. Klein, H. Koch, M. Locker, G. Lutz, H. Matthay, L. Struders, M. Trimpl, J. Ulrici, and N. Wermes, "The DEPFET pixel BIOSCOPE," *IEEE Transactions on Nuclear Science*, vol. 47, no. 3, pp. 1246–1250, 2000. [Online]. Available: https://ieeexplore.ieee.org/document/856581
(3 citations on 3 pages: 2, 3, and 10)

[46] T. Theussl, T. Moller, and M. E. Groller, "Optimal Regular Volume Sampling," in *Proceedings Visualization, 2001. VIS '01.*, 2001, pp. 91–98. [Online]. Available: https://ieeexplore.ieee.org/document/964498
(2 citations on 2 pages: 2 and 10)

[47] J. Ke, H. Yang, H. Zheng, X. Chen, Y. Jia, P. Gong, and J. Ye, "Hexagon-Based Convolutional Neural Network for Supply-Demand Forecasting of Ride-Sourcing Services," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–14, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8566163
(3 citations on 3 pages: 2, 3, and 13)

[48] M. Erdmann, J. Glombitza, and D. Walz, "A deep learning-based reconstruction of cosmic ray-induced air showers," *Astroparticle Physics*, vol. 97, pp. 46–53, 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0927650517302219
(4 citations on 3 pages: 2, 5, and 10)

[49] I. Shilon, M. Kraus, M. Büchele, K. Egberts, T. Fischer, T. L. Holch, T. Lohse, U. Schwanke, C. Steppa, and S. Funka, "Application of Deep Learning methods to analysis of Imaging Atmospheric Cherenkov Telescopes data," *Astroparticle Physics*, vol. 105, pp. 44–53, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0927650518301178
(5 citations on 3 pages: 2, 5, and 10)

[50] E. Hoogeboom, J. W. T. Peters, T. S. Cohen, and M. Welling, "HexaConv," in *International Conference on Learning Representations*, 2018. [Online]. Available: https://arxiv.org/abs/1803.02108
(3 citations on 3 pages: 2, 9, and 10)

[51] C. Steppa and T. L. Holch, "HexagDLy—Processing hexagonally sampled data with CNNs in PyTorch," *SoftwareX*, vol. 9, pp. 193–198, 2019. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352711018302723
(3 citations on 3 pages: 2, 9, and 10)

[52] T. Schlosser, M. Friedrich, and D. Kowerko, "Hexagonal Image Processing in the Context of Machine Learning: Conception of a Biologically Inspired Hexagonal Deep Learning Framework," in *2019 18th IEEE International Conference on Machine Learning and Applications (ICMLA)*, 2019, pp. 1866–1873. [Online]. Available: https://ieeexplore.ieee.org/document/8999309
(13 citations on 6 pages: 2, 3, 9, 10, 12, and 13)

[53] T. Schlosser, F. Beuth, and D. Kowerko, "Biologically Inspired Hexagonal Deep Learning for Hexagonal Image Generation," in *2020 27th IEEE International Conference on Image Processing (ICIP)*, 2020, pp. 848–852. [Online]. Available: https://ieeexplore.ieee.org/document/9190995
(5 citations on 4 pages: 2, 9, 12, and 13)

[54] T. Cevik, N. Cevik, J. Rasheed, T. Asuroglu, S. Alsubai, and M. Turan, "Reversible Logic-Based Hexel Value Differencing—A Spatial Domain Steganography Method for Hexagonal Image Processing," *IEEE Access*, vol. 11, pp. 118 186–118 203, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10296866
(2 citations on 2 pages: 2 and 13)

[55] T. Cevik, N. Cevik, J. Rasheed, A. M. Abu-Mahfouz, and O. Osman, "Facial recognition in hexagonal domain—A frontier approach," *IEEE Access*, vol. 11, pp. 46 577–46 591, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/10122525
(2 citations on 2 pages: 2 and 13)

[56] X. Zheng, "Image Segmentation for Hexagonally Sampled Images Using Statistical Region Merging," in *2022 6th International Conference on Imaging, Signal Processing and Communications (ICISPC)*, 2022, pp. 37–42. [Online]. Available: https://ieeexplore.ieee.org/document/10040503
(2 citations on 2 pages: 2 and 13)

[57] B. Gardiner, H. Zhang, M. Fayemiwo, and X. Ding, "Enhancing Medical Image Segmentation using Hexagonal Convolutional Neural Networks," in *International Conference on Artificial Intelligence in Medicine*, 2024, p. 10. [Online]. Available: https://pure.ulster.ac.uk/en/publications/enhancing-medical-image-segmentation-using-hexagonal-convolutiona
(2 citations on 2 pages: 2 and 3)

[58] Q. Liang, J. Zhou, J. Ben, Y. Chen, X. Huang, J. Ding, and J. Dai, "Precise hexagonal pixel modeling and an easy-sharing storage scheme for remote sensing images based on discrete global grid system," *International Journal of Digital Earth*, vol. 17, no. 1, p. 2328824, 2024. [Online]. Available: https://www.tandfonline.com/doi/full/10.1080/17538947.2024.2328824
(2 citations on 2 pages: 3 and 13)

[59] P. Varghese and G. Arockia Selva Saroja, "Biologically Motivated Hexagonal Framework for Machine Vision," *Lecture Notes in Networks and Systems*, vol. 435, pp. 279–290, 2022. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-19-0976-4_23
(4 citations on 3 pages: 3, 4, and 13)

[60] P. Varghese and G. A. S. Saroja, "Biologically Inspired Hexagonal Image Structure for Computer Vision," in *Advances in Information Communication Technology and Computing*, 2022, pp. 487–496. [Online]. Available: https://link.springer.com/chapter/10.1007/978-981-19-0619-0_43
(1 citation on 1 page: 3)

[61] S. Fadaei and A. Rashno, "A Framework for Hexagonal Image Processing Using Hexagonal Pixel-Perfect Approximations in Subpixel Resolution," *IEEE Transactions on Image Processing*, vol. 30, pp. 4555–4570, 2021. [Online]. Available: https://ieeexplore.ieee.org/document/9409677
(4 citations on 3 pages: 3, 9, and 12)

[62] B. Gardiner, S. Coleman, and B. W. Scotney, "Comparing Hexagonal Image Resampling Techniques with Respect to Feature Extraction," in *14th International Machine Vision and Image Processing Conference*, 2011, pp. 102–115. [Online]. Available: https://pure.ulster.ac.uk/en/publications/comparing-hexagonal-image-resampling-techniques-with-respect-to-f-3
(1 citation on 1 page: 3)

[63] M. Firouzi, S. Fadaei, and A. Rashno, "A New Framework for Canny Edge Detector in Hexagonal Lattice," *International Journal of Engineering*, vol. 35, no. 8, pp. 1588–1598, 2022. [Online]. Available: https://www.ije.ir/article_148876.html
(1 citation on 1 page: 3)

[64] N. P. Peri Hartman and S. L. Tanimoto, "A Hexagonal Pyramid Data Structure for Image Processing," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-14, no. 2, pp. 247–256, 1984. [Online]. Available: https://ieeexplore.ieee.org/document/6313207
(1 citation on 1 page: 4)

[65] S. Coleman, B. Scotney, and B. Gardiner, "Tri-directional gradient operators for hexagonal image processing," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 614–626, 2016. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S1047320316300347
(1 citation on 1 page: 4)

[66] S. A. Coleman, B. W. Scotney, and B. Gardiner, "Fast low-level multi-scale feature extraction for hexagonal images," *Proceedings of the 15th IAPR International Conference on Machine Vision Applications, MVA 2017*, pp. 342–345, 2017. [Online]. Available: https://ieeexplore.ieee.org/document/7986871
(1 citation on 1 page: 4)

[67] N. Asuni and A. Giachetti, "TESTIMAGES: A Large Data Archive For Display and Algorithm Testing," *Journal of Graphics Tools*, vol. 17, no. 4, pp. 113–125, 2013. [Online]. Available: https://www.tandfonline.com/doi/abs/10.1080/2165347X.2015.1024298
(1 citation on 1 page: 4)

[68] ——, "TESTIMAGES: a large-scale archive for testing visual devices and basic image processing algorithms," in *STAG: Smart Tools & Apps for Graphics*, 2014, pp. 63–70. [Online]. Available: https://diglib.eg.org/bitstream/handle/10.2312/stag.20141242.063-070/063-070.pdf
(1 citation on 1 page: 4)

[69] A. Meurer, C. P. Smith, M. Paprocki, O. Čertík, S. B. Kirpichev, M. Rocklin, A. Kumar, S. Ivanov, J. K. Moore, S. Singh, and others, "SymPy: symbolic computing in Python," *PeerJ Computer Science*, vol. 3, 2017. [Online]. Available: https://peerj.com/articles/cs-103/
(2 citations on 2 pages: 4 and 8)

[70] M. Linhoff, S. Bhattacharyya, J. Pérez Romero, S. Stanič, V. Vodeb, S. Vorobiov, D. Zavrtanik, M. Zavrtanik, and M. Živec, "ctapipe – Prototype Open Event Reconstruction Pipeline for the Cherenkov Telescope Array," in *Proceedings, 38th International Cosmic Ray Conference*, 2023, pp. 1–14. [Online]. Available: https://pos.sissa.it/444/703
(2 citations on 2 pages: 5 and 8)

[71] A. Azzalini and A. Capitanio, "Statistical applications of the multivariate skew normal distribution," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 61, no. 3, pp. 579–602, 1999. [Online]. Available: https://rss.onlinelibrary.wiley.com/doi/abs/10.1111/1467-9868.00194
(2 citations on 2 pages: 5 and 6)

[72] G. Vacanti, P. Fleury, Y. Jiang, E. Paré, A. C. Rovero, X. Sarazin, M. Urban, and T. C. Weekes, "Muon ring images with an atmospheric Čerenkov telescope," *Astroparticle Physics*, vol. 2, no. 1, pp. 1–11, 1994. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/0927650594900124
(2 citations on 2 pages: 5 and 6)

[73] M. Gaug, S. Fegan, A. M. W. Mitchell, M. C. Maccarone, T. Mineo, and A. Okumura, "Using Muon Rings for the Calibration of the Cherenkov Telescope Array: A Systematic Review of the Method and Its Potential Accuracy," *The Astrophysical Journal Supplement Series*, vol. 243, no. 1, p. 11, 2019. [Online]. Available: https://iopscience.iop.org/article/10.3847/1538-4365/ab2123/meta
(2 citations on 2 pages: 5 and 6)

[74] M. d. Naurois, "A high performance likelihood reconstruction of gamma-rays for Imaging Atmospheric Cherenkov Telescopes," *Astroparticle Physics*, vol. 32, no. 5, pp. 231–252, 2009. [Online]. Available: https://www.sciencedirect.com/science/article/abs/pii/S0927650509001364
(2 citations on 2 pages: 5 and 6)

[75] A. Krizhevsky, "Learning Multiple Layers of Features from Tiny Images," University of Toronto, Tech. Rep., 2009. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.222.9220&rep=rep1&type=pdf
(6 citations on 4 pages: 6, 7, 10, and 12)

[76] L. N. Darlow, E. J. Crowley, A. Antoniou, and A. J. Storkey, "CINIC-10 Is Not ImageNet or CIFAR-10," University of Edinburgh, Tech. Rep., 2018. [Online]. Available: https://arxiv.org/abs/1810.03505
(4 citations on 4 pages: 6, 7, 10, and 12)

[77] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia Object Image Library (COIL-20)," Columbia University, Tech. Rep., 1996. [Online]. Available: http://www.cs.columbia.edu/CAVE/publications/pdfs/Nene_TR96.pdf
(4 citations on 4 pages: 6, 8, 10, and 12)

[78] ——, "Columbia Object Image Library (COIL-100)," Columbia University, Tech. Rep., 1996. [Online]. Available: https://citeseerx.ist.psu.edu/doc/10.1.1.360.6420
(4 citations on 4 pages: 6, 8, 10, and 12)

[79] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-Based Learning Applied to Document Recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998. [Online]. Available: http://yann.lecun.com/exdb/publis/pdf/lecun-01a.pdf
(4 citations on 4 pages: 6, 8, 10, and 12)

[80] Y. Le and X. Yang, "Tiny ImageNet Visual Recognition Challenge," Stanford University, Tech. Rep., 2015. [Online]. Available: http://cs231n.stanford.edu/reports/2015/pdfs/yle_project.pdf
(4 citations on 4 pages: 6, 8, 10, and 12)

[81] T. P. Robitaille, E. J. Tollerud, P. Greenfield, M. Droettboom, E. Bray, T. Aldcroft, M. Davis, A. Ginsburg, A. M. Price-Whelan, W. E. Kerzendorf, and others, "Astropy: A community Python package for astronomy," *Astronomy & Astrophysics*, vol. 558, no. A&A, pp. 1–9, 2013. [Online]. Available: https://www.aanda.org/articles/aa/full_html/2013/10/aa22068-13/aa22068-13.html
(1 citation on 1 page: 8)

[82] K. Bernlöhr, "Simulation of imaging atmospheric Cherenkov telescopes with CORSIKA and sim_telarray," *Astroparticle Physics*, vol. 30, no. 3, pp. 149–158, 2008. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0927650508000972
(1 citation on 1 page: 8)

[83] K. He, X. Zhang, S. Ren, and J. Sun, "Deep Residual Learning for Image Recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 770–778. [Online]. Available: https://ieeexplore.ieee.org/document/7780459
(2 citations on 2 pages: 9 and 10)

[84] ——, "Identity Mappings in Deep Residual Networks," in *European Conference on Computer Vision*, 2016, pp. 630–645. [Online]. Available: https://link.springer.com/chapter/10.1007/978-3-319-46493-0_38
(2 citations on 2 pages: 9 and 10)

[85] T. Schlosser, M. Friedrich, T. Meyer, and D. Kowerko, "A Consolidated Overview of Evaluation and Performance Metrics for Machine Learning and Computer Vision," 2024. [Online]. Available: https://www.researchgate.net/publication/374558675_A_Consolidated_Overview_of_Evaluation_and_Performance_Metrics_for_Machine_Learning_and_Computer_Vision
(1 citation on 1 page: 9)

[86] D. J. Field, "Relations between the statistics of natural images and the response properties of cortical cells," *Journal of the Optical Society of America A*, vol. 4, no. 12, pp. 2379–2394, 1987. [Online]. Available: https://www.osapublishing.org/josaa/abstract.cfm?uri=josaa-4-12-2379
(1 citation on 1 page: 9)

```
1   # Example "line": nearest neighbor interpol. with colormap "gray"
2   plot_function_and_visualize_hexarray(
3       # Function plot_function()
4       plot_function_function = plot_function_hexagonal,
5       function_s          = ['x'],
6       symbol_s            = ['x'],
7       figure_size         = 60,
8       window_size         = 1,
9       step_size           = 0.005,
10      linewidth_factor    = 0.1,
11      rad_o               = 0.63,
12      rotation_degrees    = 0,
13      interpolation       = 'nearest neighbor',
14      output_dir          = output_dir,
15      # Function visualize_hexarray()
16      colormap        = 'gray',
17      visualize_axes = True,
18      show_hexarray   = True)
```

Listing 3: Geometric primitive image generation example 1.

(a) With image resolution axes   (b) Without axes
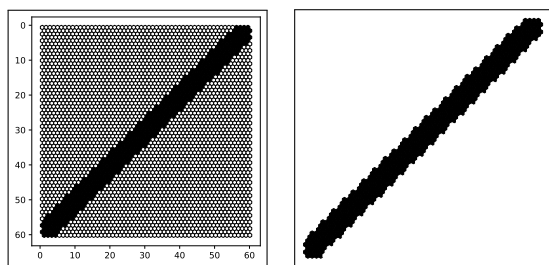
Figure 21: Geometric primitive image generation example 1.

```
1   # Example "curve": nearest neighbor interpol. with colormap "gray"
2   plot_function_and_visualize_hexarray(
3       # Function plot_function()
4       plot_function_function = plot_function_hexagonal,
5       function_s          = ['sqrt(x)'],
6       symbol_s            = ['x'],
7       figure_size         = 60,
8       window_size         = 1,
9       step_size           = 0.005,
10      linewidth_factor    = 0.1,
11      rad_o               = 0.63,
12      rotation_degrees    = 0,
13      interpolation       = 'nearest neighbor',
14      output_dir          = output_dir,
15      # Function visualize_hexarray()
16      colormap        = 'gray',
17      visualize_axes = True,
18      show_hexarray   = True)
```

Listing 4: Geometric primitive image generation example 2.

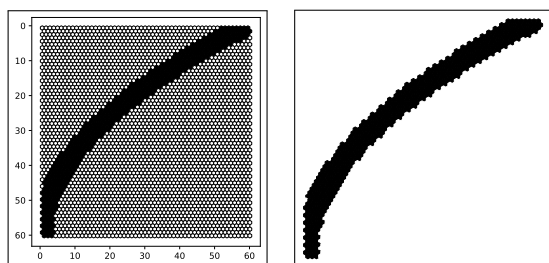(a) With image resolution axes   (b) Without axes

Figure 22: Geometric primitive image generation example 2.

```
1   # Example "circle": linear interpol. with colormap "inferno"
2   plot_function_and_visualize_hexarray(
3       # Function plot_function()
4       plot_function_function = plot_function_hexagonal,
5       function_s          = ['1/4*(2-sqrt(-3+16*x-16*x^2))',
6                             '1/4*(2+sqrt(-3+16*x-16*x^2))'],
7       symbol_s            = ['x'],
8       figure_size         = 60,
9       window_size         = 1,
10      step_size           = 0.005,
11      linewidth_factor    = 0.2,
12      rad_o               = 0.63,
13      rotation_degrees    = 0,
14      interpolation       = 'linear',
15      output_dir          = output_dir,
16      # Function visualize_hexarray()
17      colormap        = 'inferno',
18      visualize_axes = True,
19      show_hexarray   = True)
```
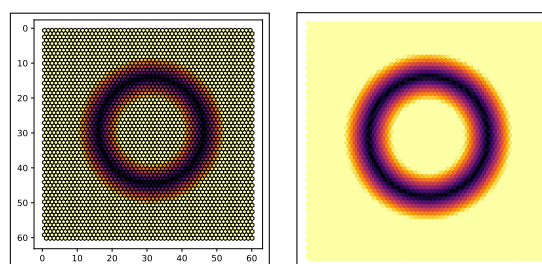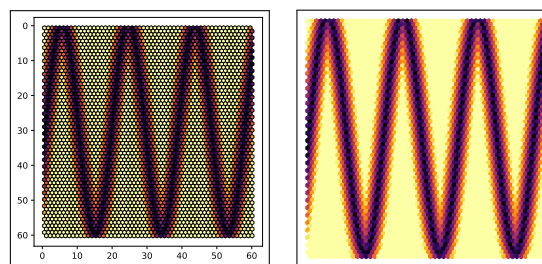
Listing 5: Geometric primitive image generation example 3.

(a) With image resolution axes   (b) Without axes

Figure 23: Geometric primitive image generation example 3.

```
1   # Example "sine": linear interpol. with colormap "inferno"
2   plot_function_and_visualize_hexarray(
3       # Function plot_function()
4       plot_function_function = plot_function_hexagonal,
5       function_s          = ['0.5*sin(20*x)+0.5'],
6       symbol_s            = ['x'],
7       figure_size         = 60,
8       window_size         = 1,
9       step_size           = 0.005,
10      linewidth_factor    = 0.1,
11      rad_o               = 0.63,
12      rotation_degrees    = 0,
13      interpolation       = 'linear',
14      output_dir          = output_dir,
15      # Function visualize_hexarray()
16      colormap        = 'inferno',
17      visualize_axes = True,
18      show_hexarray   = True)
```

Listing 6: Geometric primitive image generation example 4.

(a) With image resolution axes   (b) Without axes

Figure 24: Geometric primitive image generation example 4.

(a) 60 × 60   (b) 80 × 80   (c) 100 × 100   (d) 200 × 200
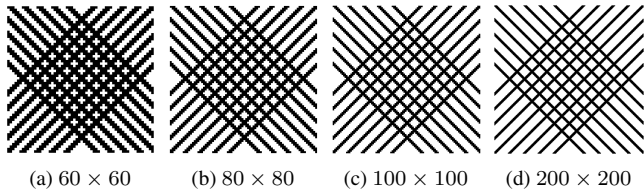
Figure 25: Geometric primitive images in ascending figure size: square.

(a) 0.001   (b) 0.01   (c) 0.03   (d) 0.05

Figure 26: Geometric primitive images in ascending step size: square.

(a) 0.005   (b) 0.01   (c) 0.02   (d) 0.04

Figure 27: Geometric primitive images in ascending linewidth: square.

(a) 0   (b) 30   (c) 60   (d) 90

Figure 28: Geometric primitive images in ascending rotation degree: square.

(a) 0.75   (b) 1.0   (c) 1.25   (d) 1.5

Figure 29: Geometric primitive images in ascending function factor: square.

(a) 60 × 60   (b) 80 × 80   (c) 100 × 100   (d) 200 × 200

Figure 30: Geometric primitive images in ascending figure size: hexagonal.

(a) 0.001   (b) 0.01   (c) 0.03   (d) 0.05

Figure 31: Geometric primitive images in ascending step size: hexagonal.

(a) 0.005   (b) 0.01   (c) 0.02   (d) 0.04

Figure 32: Geometric primitive images in ascending linewidth: hexagonal.

(a) 0   (b) 30   (c) 60   (d) 90

Figure 33: Geometric primitive images in ascending rotation degree: hexagonal.

(a) 0.75   (b) 1.0   (c) 1.25   (d) 1.5

Figure 34: Geometric primitive images in ascending function factor: hexagonal.

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$

(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$
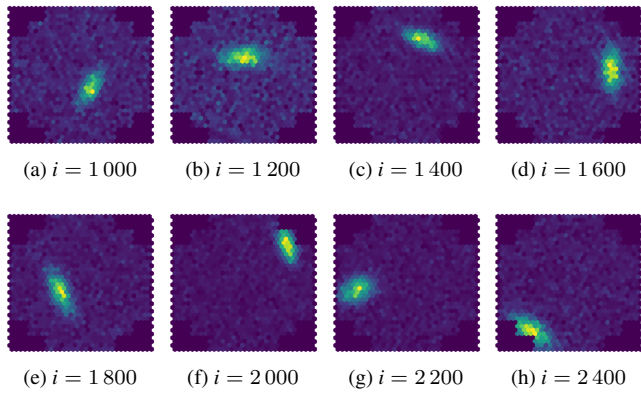
Figure 35: Astronomical image generation images in ascending intensity (i): single shower area, Gaussian.

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$

(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$
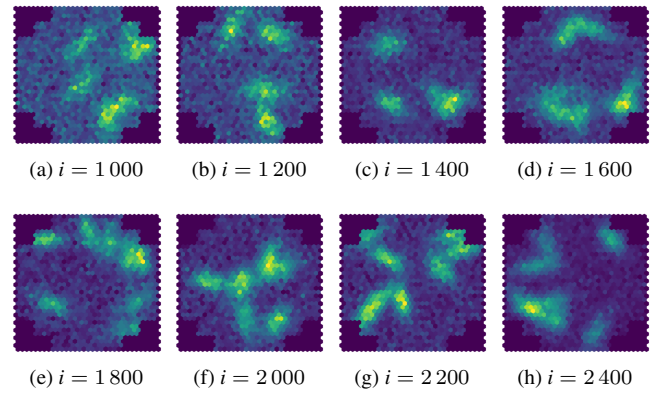
Figure 38: Astronomical image generation images in ascending intensity (i): multiple shower areas, Gaussian.

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$
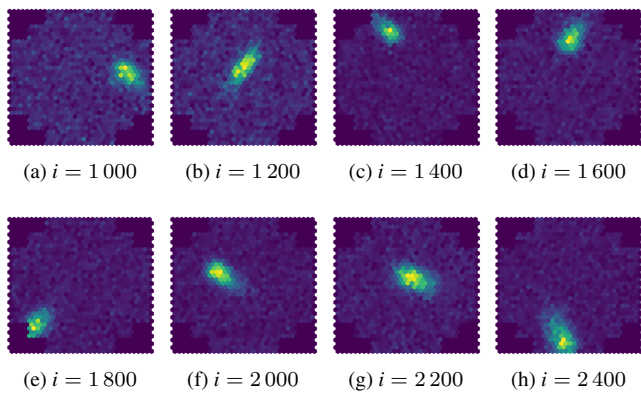
(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$

Figure 36: Astronomical image generation images in ascending intensity (i): single shower area, skewed Gaussian.

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$

(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$
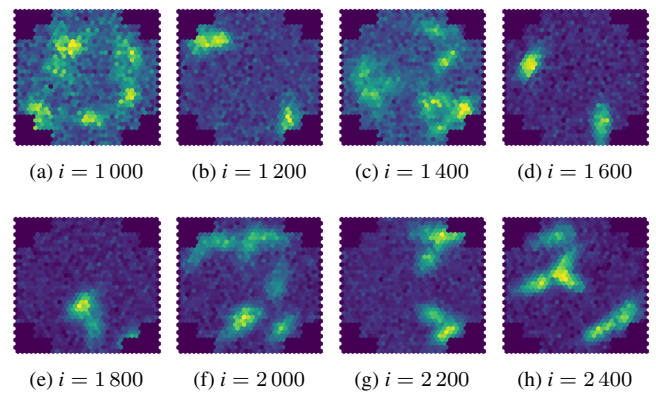
Figure 39: Astronomical image generation images in ascending intensity (i): multiple shower areas, skewed Gaussian.
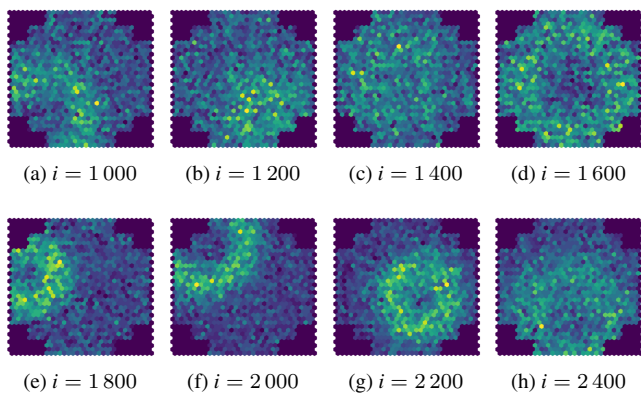

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$

(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$

Figure 37: Astronomical image generation images in ascending intensity (i): single shower area, ring Gaussian.

(a) $i = 1\,000$ (b) $i = 1\,200$ (c) $i = 1\,400$ (d) $i = 1\,600$

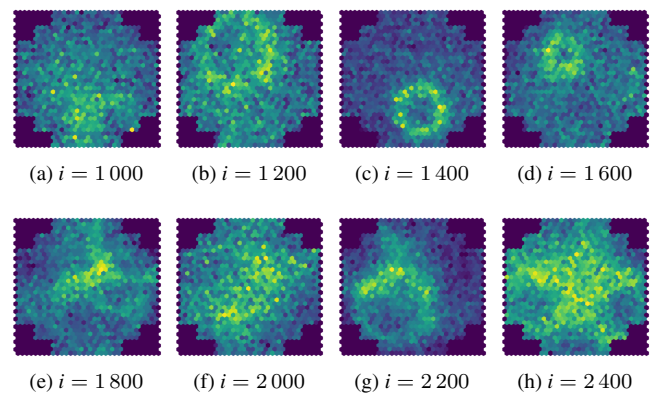(e) $i = 1\,800$ (f) $i = 2\,000$ (g) $i = 2\,200$ (h) $i = 2\,400$

Figure 40: Astronomical image generation images in ascending intensity (i): multiple shower areas, ring Gaussian.

TOBIAS SCHLOSSER received his B.Sc., M.Sc., and Ph.D. degrees in Computer Science and Artificial Intelligence from the Chemnitz University of Technology, Germany in 2016, 2018, and 2024, respectively. His research interests lie in the fields of Computer Vision and Pattern Recognition, Image Processing, as well as Machine Learning and Deep Learning, including application areas in the domains of Medical and Industrial Text, Image, and Video Processing. His main research focus lies in the synthesis of biologically inspired hexagonal image processing with approaches and models from deep learning.

MICHAEL FRIEDRICH Received his M.Sc. degree in Applied Computer Science from the Chemnitz University of Technology, Germany in 2022. He is currently a research fellow in the fields of Computer Science and Artificial Intelligence with a focus on Computer Vision, Pattern Recognition, Image Processing, as well as Machine Learning and Deep Learning.

TRIXY MEYER studied Biomedical Engineering at the Chemnitz University of Technology, Germany. She received her B.Sc. and M.Sc. degrees in 2018 and 2021, respectively. Her main research focus lies with the processing and visualization of medical data, especially medical texts, utilizing approaches from machine learning as well as natural language processing.

MAXIMILIAN EIBL studied Information Science, History, and Philosophy as well as Computer Science at the University of Regensburg, the University of Illinois Urbana-Champaign, the University of Sheffield, and the University of Koblenz and Landau from 1989 to 2000. Currently, Maximilian Eibl is the head of the Professorship of Media Informatics at the Chemnitz University of Technology, Germany. He is also the Vice Rector for Education at the Chemnitz University of Technology. His research interests include human computer interaction, multimedia information retrieval, digital learning, and media distribution.

DANNY KOWERKO received his diploma in physics and his Ph.D. degree at the Department of Natural Sciences of the Chemnitz University of Technology in 2005 and 2010, respectively. After a postdoc at the University of Zurich in Switzerland from 2011–2015 with a biocomputational focus, he returned to the Chemnitz University of Technology and became a junior professor in the Department of Computer Science in 2019. His junior professorship *Media Computing* is dedicated to algorithm and software development in the fields of computer vision, biocomputation, computational ophthalmology, hexagonal image processing, and general object detection and classification.

• • •