applied optics

Algorithm for detecting airborne objects with a thermal infrared camera to ensure a safe operation of laser-optical ground stations

Jakob Steurer,[1] Nils Bartels,[1,*] Daniel Hampf,[2] Felicitas Niebler,[1] Tristan Meyer,[1] Wolfgang Riede,[1] and Thomas Dekorsy[1]

[1]*German Aerospace Center (DLR), Institute of Technical Physics, Pfaffenwaldring 38-40, 70569 Stuttgart, Germany*
[2]*DiGOS Potsdam GmbH, Telegrafenberg 1, 14473 Potsdam, Germany*
**nils.bartels@dlr.de*

Received 6 May 2024; revised 25 July 2024; accepted 26 July 2024; posted 26 July 2024; published 13 August 2024

Laser-optical ground stations play an important role for satellite laser communication and satellite laser ranging (SLR). The safe operation of lasers in public airspace, which usually requires approval by legal entities, requires reliable, redundant, and independent systems for airborne object detection to avoid a potentially hazardous laser exposition. In this work, we propose an algorithm based on classical image filtering and thresholding to detect aircraft in images taken with a thermal infrared camera. The algorithm is optimized and evaluated with an image dataset acquired by the infrared camera mounted to SLR station miniSLR located in Stuttgart, Germany. Despite its simplicity and efficiency (7 ms for an image with 640 × 512 pixels on a standard consumer PCU), we find that the proposed algorithm has a high accuracy, yielding a 99.8% correct classification of images. Although laser safety systems require several independent aircraft detection methods, the proposed algorithm might be a valuable contribution for companies and institutes with the need to operate lasers in public airspace.

https://doi.org/10.1364/AO.529222

1. INTRODUCTION

Laser-optical ground stations have many applications in science and technology. Particularly, high-precision laser-based distance measurement to satellites via satellite laser ranging (SLR) [1] is a key technology of geodesy, for example, to establish and maintain the International Terrestrial Reference Frame (ITRF), to determine Earth rotation parameters, or to characterize the movement of tectonic plates. Similarly, lunar laser ranging allows one to measure the precise distance to retroreflector arrays placed on the moon [2]. Using large telescopes and intense lasers, it is also possible to measure distance to space objects without retroreflectors (e.g., space debris), which is known as space debris laser ranging [3]. Scientists working in this field are organized in the International Satellite Laser Ranging Service (ILRS) [2]. Furthermore, laser-optical ground stations are also used in ground-to-satellite laser communication [4,5]. Compared to the more common way of communication using radio waves, lasers have the advantage of an increased range (due to achievable low laser beam divergence), as well as a high bandwidth. Additionally, the low beam spread of lasers provides inherent safety against eavesdropping. Laser-optical ground stations may also be used in areas such as quantum-key distribution [6], laser time transfer [7,8], laser momentum transfer (for collision avoidance in space) [9], or laser-ablative removal of space debris [10–12]. Finally, lasers are also used in outdoor laser shows [13]. In many cases, it is not feasible to operate laser-optical ground stations with eye-safe laser parameters. For SLR and laser communication this is mostly due to limitations from the link budget and the need to achieve a sufficient signal to noise ratio. Thus, laser-optical ground stations need reliable procedures and safety equipment to ensure laser safety on ground (e.g., for personnel operating the SLR station), but additionally also in the (often public) airspace around the ground station. Thus, in addition to the common methods to secure ground personnel, such as laser safety glasses, door interlocks, restricted access areas, laser safety barriers, laser safety training, etc., such ground stations also require systems to reliably detect airborne objects such as airplanes in order to interrupt the laser emission (e.g., via laser shutters, by using the laser interlock or by interrupting the trigger signal of externally triggered lasers) in time to avoid a potentially hazardous laser exposition [14]. To enhance the overall detection rate and thus the laser safety, optical ground stations typically ensure laser safety using a combination of different detection systems for airborne objects. These sensors should operate independently and the detection of an airborne object by any sensor leads to an interruption

Table 1. Overview of Commonly Used Sensors for Aircraft Detection for Laser Safety[a]

Sensor Category	Sensor Type	Application	Limitation	References
Human	Human spotter	All (large) objects in the visible	Limited human attention	
Receivers	Mode-S, ADS-B, MLAT	All aircrafts using transponders	Transponders needed	[15–17]
Radars	Active radar	Objects in range of radar stations	Expensive, but sometimes available as air traffic data	[18–20]
Acoustic	Microphone array	Low flying aircraft	Characteristic acoustic signal needed	[21]
Visible camera	All-sky or in-beam camera	Aircraft in field of view	Requires optical contrast between aircraft and background	[22]
Infrared camera	All-sky or in-beam camera	Aircraft in field of view	Requires temperature contrast between aircraft and background	[23–26]

[a] ADS-B = automatic dependent surveillance-broadcast, MLAT = multilateralization.

of laser emission. Table 1 provides an overview of commonly used sensors for detecting airborne objects in the vicinity of laser-optical ground stations.

Main reasons why camera systems are often used as part of the aircraft detection system are

- the redundancy in detection (independent of other sensors);
- relatively affordable sensors (~$US 10, 000);
- the possibility to detect rare and exceptional airborne objects such as sky-divers or hot-air balloons that might not always be detected by radars or emit a transponder signal.

In order to significantly contribute to the overall aircraft safety for laser-optical ground stations, the camera-based detection needs to be optimized and carefully tested both in terms of hardware (e.g., selection of wavelength range, field of view, etc.), as well as in terms of software. Here, a major challenge is finding appropriate algorithms for airborne object detection. In this work, we share results from a recent detection and evaluation campaign of airborne objects with a thermal infrared camera mounted to our SLR station miniSLR. Based on this work, we introduce a specific algorithm for the classification of images that has produced good results in our test campaign. The algorithm is described in detail and the underlying thermal infrared images are available in a public repository to enhance the open exchange of data and knowledge in order to further improve safety in public airspace.

2. MATERIALS AND METHODS

Figure 1 shows an annotated image of the SLR station miniSLR [27] that has been built by and is operated at the Institute of Technical Physics at DLR in Stuttgart, Germany, with a focus on components relevant for laser safety.

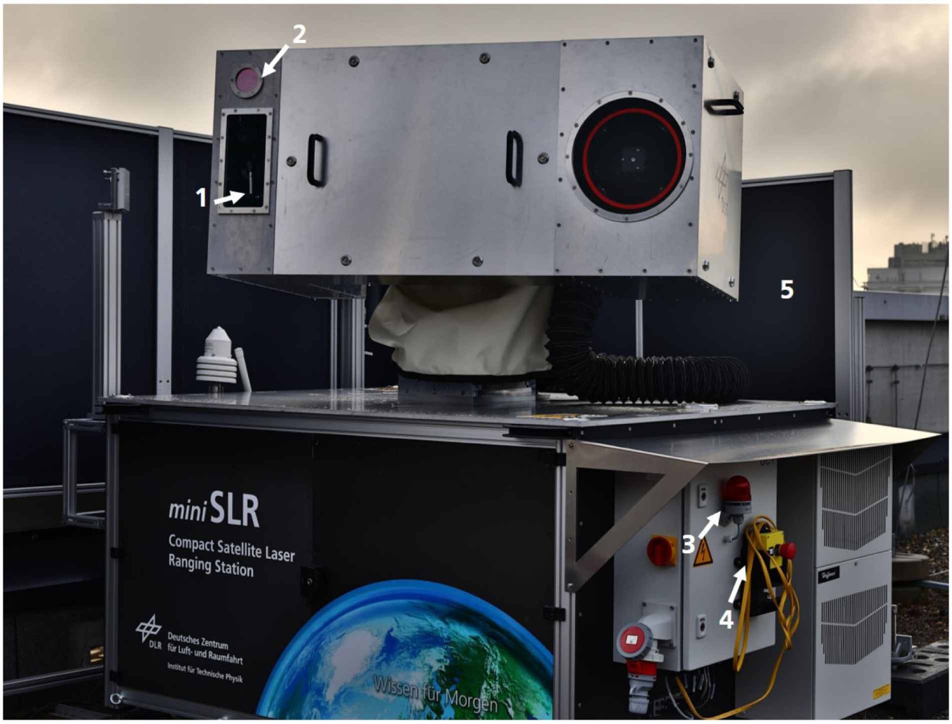
2
1
5
miniSLR
Compact Satellite Laser Ranging Station
Deutsches Zentrum für Luft- und Raumfahrt
Institut für Technische Physik
Wissen für Morgen
3
4

Fig. 1. Annotated image of the miniSLR, showing key components of the laser safety system with the laser transmitter window (1), the Germanium window of the thermal infrared camera (2), laser warning lamp (3), emergency stop button (4), and physical laser safety barriers (5).

Table 2. Technical Specifications of the Infrared Camera[a]

Specification	Parameter
Image resolution	640 × 512 pixels
Focal length	60 mm
Field of view	10.4 × 8.3 deg
Spectral range	7.5–13.5 μm
Pixel size	17 μm
Thermal sensitivity	<50 mK

[a] Tau2, Teledyne FLIR LLC.

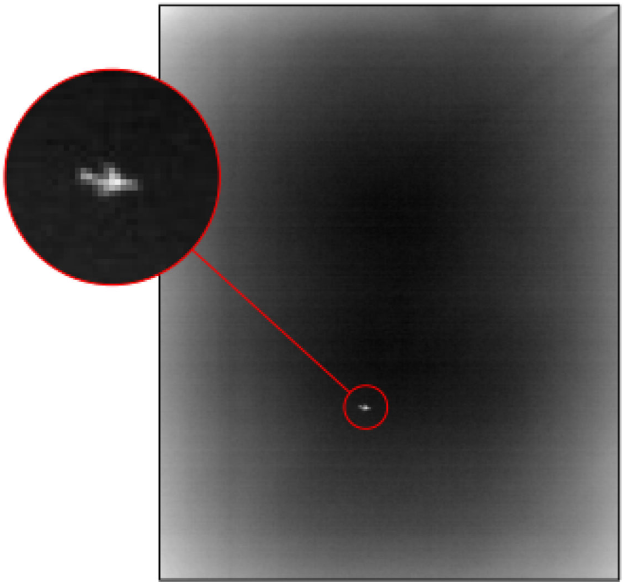

Fig. 2. Example image taken with the thermal infrared camera showing a close airplane in front of a clear sky. The part shown in the red circle is magnified by a factor of five.

Specifications of the infrared camera used for aircraft detection (Tau2, Teledyne FLIR LLC) are provided in Table 2. Images are provided as 16-bit gray-scale images.

The camera is mounted to the movable platform of the miniSLR (behind a Germanium window; see label "2" in Fig. 1) and thus always points in the direction of laser emission. The camera is most sensitive in the range from 7.5 to 13.5 μm and thus peaks for objects with a temperature between −60°C and 115°C emitting thermal radiation. Note that this long wave infrared radiation will also have a high transmission through the atmosphere (higher than light in the visible). Furthermore, the visibility of objects is independent of the sun position (as opposed to visible images) and any hot components of an aircraft (e.g., engines) will be very prominent in images [24]. Figure 2 shows an example of an image taken with the camera showing a detected aircraft in front of a clear sky. It becomes clear that the airplane is nicely visible, despite covering only a few pixels in the image.

Note that the size of an object x_{object} (e.g., an aircraft) required to fill a pixel on the camera depends on its distance and is described by [28]

$$x_{\mathrm{object}} = \frac{x_{\mathrm{pix}} d}{f}, \tag{1}$$

where x_{pix} is the pixel size of the camera sensor, d is the range to the aircraft, and f is the focal length of the camera.

With the described configuration, a single pixel corresponds to 3 cm object size at 100 m distance, to 28 cm object size at 1 km and to 2.8 m at 10 km distance. For the laser parameters of the miniSLR (85 μJ laser pulse energy, 50 kHz pulse repetition rate, 50 μrad beam divergence, 1064 nm wavelength, 5 cm exit beam diameter, 500 ps pulse duration) [27], any objects at a distance greater than 10 km are not threatened, since this is above the no-hazard distance ("NOD") of the laser as calculated by international laser safety regulations. This means that even small aircraft (typical wingspan at least 8 m) and helicopters (typical cabin length 4 m) will cover more than one pixel. The camera is used with a short exposure time of 3 ms using a global, electronic shuttering mode. This means that sharp images of moving aircraft are obtained even when the mount of the SLR station (and thus the camera) is rotating, which can have a slewing rate as fast as 1°/second.

Using the described sensor, we measured a set of 97,514 images covering different weather conditions (with and without clouds) and different times of the day. The images obtained by the sensor were afterwards classified by a combination of selection by hand and by specifically investigating pictures, in which an aircraft was expected due to information from the DFS (DFS stands for "Deutsche Flugsicherung", in English, German Aviation Safety) data stream (radar data). This was used to produce a dataset of 1284 images used as a benchmark, which was distributed among the following categories.

1. "Critical aircraft": aircraft with clearly observable features, e.g., signature of engine or shape of wings.
2. "Non-critical aircraft": aircraft only visible as distant dot (no features like wings or engine). Can be distinguished from birds only by the straight path (and slow speed) at which they cross the image.
3. "Clouds": images showing clouds, and no aircraft. Different scenarios with different cloud coverage.
4. "Birds": images showing flying animals (e.g., birds or bats); recognizable by shape or due to a cycling movement.
5. "Antenna": some images show a blurred antenna in the background, which is above the elevation mask of the miniSLR.
6. "Clear sky": images without clouds or aircraft.
7. "Interesting images": images that were initially incorrectly classified by hand (mostly from the "clouds" category), but where an aircraft was detected by later analysis with an image processing algorithm. These are the most challenging images showing that image processing can outperform a human spotter.

Table 3 provides an overview of the number of images for each of the defined categories. While the dataset covers 800 images with aircraft, it does unfortunately not contain any images of helicopters, air balloons, or a sky-diver. Details on the distribution of images on the time of the day and the capturing data are provided in Appendix A.

As the target for the image classification the categories "safe" (where laser emission is allowed) and "unsafe" (no laser emission allowed) are defined. Of course, laser emission is not allowed in the case of a detected aircraft. Although, aircraft from the

Table 3. Number of Images and Targeted Classification of Different Categories in the Generated Dataset

Category	Number of Images	Target Classification
Critical aircraft	359	Unsafe
Non-critical aircraft	434	Unsafe
Clouds	146	Safe
Birds	143	Unsafe
Antenna	35	Safe
Clear sky	158	Safe
Interesting images	9	Unsafe
Total	1284	–

category “non-critical aircraft” (meaning only a few pixels wide) should be at a safe distance (e.g., two pixels correspond to 5.4 m in size; see previous discussion), it was decided to classify these as unsafe to rule out the possibility of illuminating an aircraft. It was also decided to classify images with “birds” as unsafe, firstly to avoid disturbance of wildlife and secondly, because birds and distant aircraft cannot be distinguished if they only spawn a few pixels in the image. Images from the categories “clear sky”, “clouds”, and “antenna” should be considered as safe for ranging. The antenna cannot accidentally be irradiated by the laser due to a software-based elevation mask. The decision to label images with clouds for ranging is made to allow for SLR in the presence of partial cloud coverage. In general, SLR will only be performed when there are only a few clouds. Note that it is in principle possible that an aircraft will be hidden behind a cloud (where laser irradiation is strongly attenuated) and suddenly be revealed due to aircraft or cloud movement (no power attenuation). In that scenario, the laser safety should be ensured by the other aircraft sensors (e.g., the datasream of radar data from air traffic management). Furthermore, the infrared camera and image processing need to be performed at a reasonably high frame rate (currently 1 Hz at the miniSLR) to minimize the risk of unintentional irradiation of an airborne object. It is also advisable to introduce a delay time (e.g., several seconds) before the laser may be emitted again after any aircraft detection by the camera.

3. RESULTS

A. Description of the Image Processing Algorithm

To our knowledge, the only detailed description of an image processing algorithm for aircraft detection with a thermal infrared camera in the vicinity of a laser-optical ground station has been provided by Leidig *et al*. from the Geodetic Observatory in Wettzell, Germany [24]. In the described algorithm, the authors used three steps. In the first step, a median filter was applied to reduce noise in the infrared images. This median filter is a sliding window that replaces each pixel by the median of the neighboring entries. In a second step, a specific edge detector (an edge means a connected gradient in intensity) known as the Kenny algorithm was applied to generate a binary image containing only the edges. In the third step, objects were filtered in terms of a minimum size to reject seed artifacts or insects. While the authors found a detection rate of >97% for airplanes at a distance up to 10 km, they also reported false alarms on clouds and birds (note that we do not consider the detection of a bird and resulting laser shutdown as “false alarm” in our work).

For the analysis of our dataset of images, we have first investigated several different algorithms for image analysis [26]. In this article, we will only describe the best tested algorithm; see Fig. 3.

Figure 3(A) shows an example of an original thermal infrared image, which contains clouds as well as an airplane. The airplane is also shown with 5× magnification in the red circle and the goal of the algorithm is the classification as “unsafe” image. In “step 1” of the algorithm, the original image is filtered with a 5×5 median filter [Fig. 3(B)]. In our software, which uses Python 3, this is done with the library “OpenCV” using the command “medianBlur()”, which accepts 16-bit images as input for the median filter. This median filter has a high performance and uses an optimized algorithm from Perrault and Herbert [29]. In contrast to the work from Leidig *et al*. [24], airplanes are not detected from the median filtered image. Instead, the median filtered image is used as a background, meaning that a new image [(shown in Fig. 3(C)] is generated by subtracting the median filtered image from the original image (“step 2”). This idea is based on Mednieks [30], who used this concept for identifying foreign objects in X-ray images of processed food. In this subtracted image, the aircraft is already nicely visible in front of a mostly black background. Only small artifacts (fortunately less bright than the aircraft) are visible on the edges of the large cloud [see the larger red circle in Fig. 3(C), which shows an area with 5× magnification].

In “step 3”, the image is binarized using a threshold of the form

$$T = \bar{x} + k\sigma, \tag{2}$$

where $\bar{x}$ is the mean intensity and σ is the standard deviation of all pixels in the image. The parameter k needs to be optimized using acquired images to obtain a good classification. The binarized image of Fig. 3(D) used a value of $k = 12$. All pixels with an intensity above T are set to one and the other pixels are set to zero. In “step 4”, we furthermore remove single positive pixels (meaning pixels with value 1, where all eight neighboring pixels have value 0) from the image using a hit-or-miss transform. This will remove single-pixel artifacts from the image while retaining the structure of the aircraft, which will typically have at least one neighboring positive pixel. Finally, the classification (“step 5” of the algorithm) is done by checking if the image has at least one pixel with value 1, in which case it is labelled as “unsafe” and otherwise as “safe”. For the interested reader, the detailed implementation (source code) of the algorithm is presented in Appendix B.

B. Optimization of Algorithm Parameters

In the described algorithm the median filter size had already been fixed to a size of 5×5, and thus the only variable parameter that needs to be optimized for the classification is the number of standard deviations k used for the threshold. Figure 4 shows the fraction of correctly classified images (safe versus unsafe) for the different categories of our dataset as a function of the threshold parameter k. The challenge here is to find a compromise such that the algorithm will on the one hand have a high rate of aircraft detection, even for distant aircraft (category “non-critical aircraft”), while maintaining a low fraction of images from

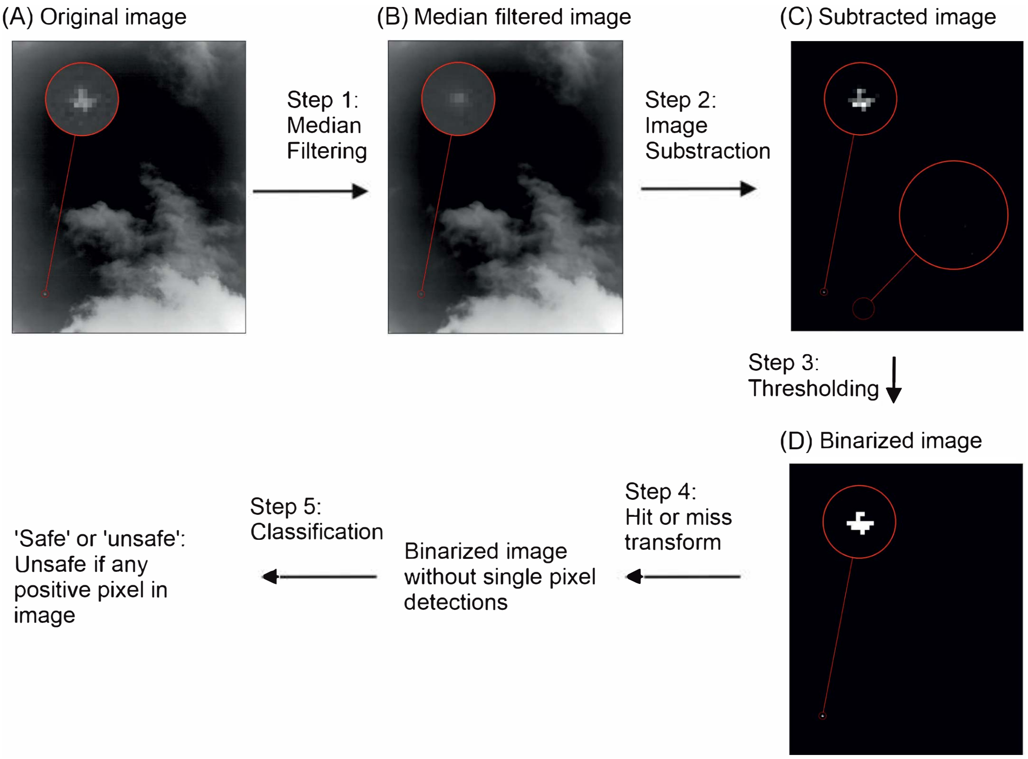
(A) Original image
(B) Median filtered image
(C) Subtracted image
Step 1: Median Filtering
Step 2: Image Substraction
Step 3: Thresholding
(D) Binarized image
Step 4: Hit or miss transform
Binarized image without single pixel detections
Step 5: Classification
'Safe' or 'unsafe': Unsafe if any positive pixel in image

Fig. 3. Schematic description of the image detection algorithm proposed in this work. The different panels and image processing steps are described in the text.

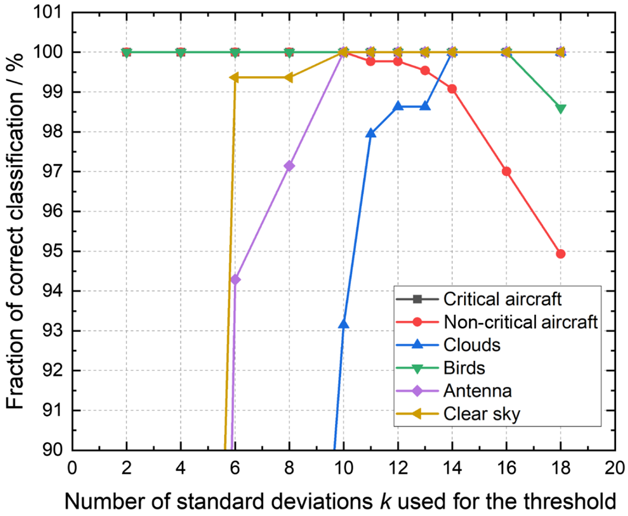

Fig. 4. Fraction of correct classifications as a function ("safe" or "unsafe") of the optimized parameter k for the different image categories.

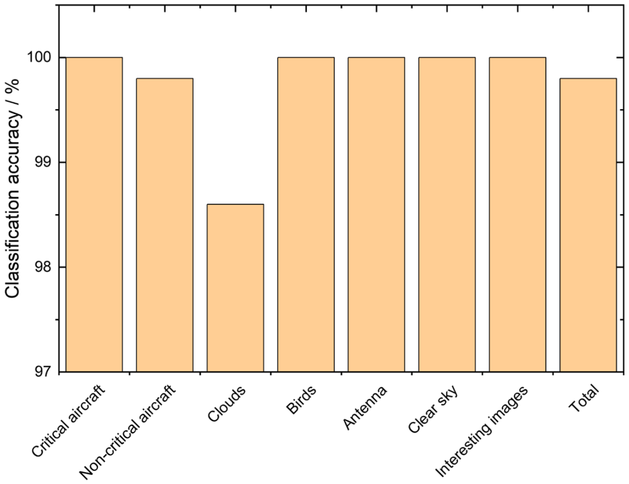

Fig. 5. Results of classification accuracy (safe/unsafe) for the proposed image processing algorithm with $k = 12$.

the cloud category as "unsafe", in order to not unnecessarily interrupt the laser emission.

For the selected optimal parameter of $k = 12$, we find that all but one image from the categories "critical aircraft" and "non-critical aircraft" are correctly classified as "safe" or "unsafe"; see Fig. 5.

Furthermore, we find that (neglecting images of the category "interesting images") a total number of 1272 out of 1275 images were correctly classified as "safe" or "unsafe". This corresponds to a detection accuracy of $1272/1275 = 99.8\%$.

C. Analysis of Processing Time

Another important parameter for the algorithm is the processing speed. The processing speed was tested on the computer of the miniSLR (AMD Ryzen 7 2700X, with 3.7 GHz frequency and 32 GB of RAM), which is a mid-range consumer PC from 2018 without dedicated GPU. Images were acquired at a frame rate of 1 Hz and the distribution of the processing time (between image acquisition and final classification) was measured over 56360 images; see Fig. 6. It was found that images were processed with a time between 4 and 12 ms. This means that the algorithm is sufficiently fast to even increase the frame rate to two or even four

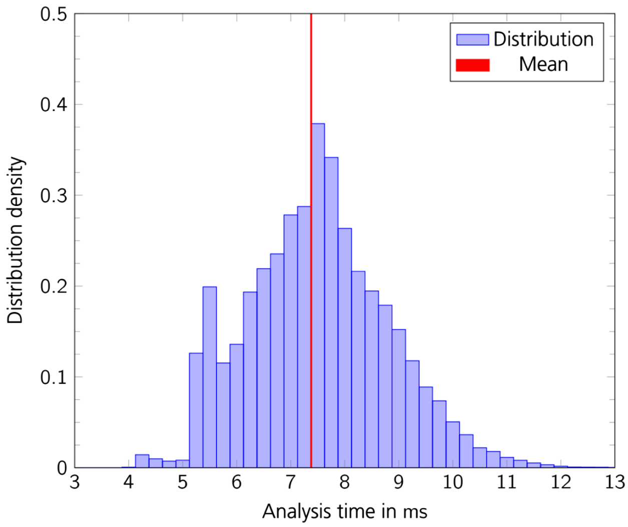

Fig. 6. Distribution of execution times for the developed image analysis algorithm on the miniSLR main computer. Image taken from Ref. [26].

images per second to enhance safety for fast-moving, low-flying aircraft [26].

4. DISCUSSION

We have generated a classified dataset of 1284 thermal infrared images and introduced a specific algorithm for aircraft detection. The algorithm shows a very high reliability in terms of the detection of aircraft (>99.8% in the images of our dataset) and only a small fraction of false-positive detections in case of cloud coverage (<1.5% of images from the "clouds" category of our dataset). The detection algorithm has a fast processing speed on a standard PC (execution time of a few milliseconds) and can likely even be implemented on a microprocessor. By connecting the camera to the microprocessor, it would be possible to build a small and affordable aircraft detection system that can then be easily integrated into many existing optical ground stations. A limitation of the presented algorithm is that it can only be used to detect aircraft in front of a sky and not in more complex scenarios such as in front of buildings or other structures. The reason for this is that the algorithm simply looks for any sharp differences in the contrast (meaning temperature differences) of the thermal infrared images and relies on the fact that these features are not present in clouds. In such more complex scenarios, aircraft detection might require modern approaches such as machine learning, which would however need a much larger dataset for training the model. An important task for future analysis would be to prove that the algorithm does not only detect airplanes and birds, but also rare events such as incoming hot-air balloons, helicopters, or sky-divers. This could so far not be tested, since these events did not occur near our SLR station and might require taking thermal infrared images in front of a sky background at other locations. Since the presented algorithm detects any sharp thermal contrasts, it is however very likely that it will also detect such events without need for adaptation. The dataset analyzed in this work also does not include any images of the moon. During the review process for this publication, we have however taken infrared images of the moon during one night (almost a full moon July 20th, 2024). These images were classified as "safe" by the algorithm.

Of course, by far the safest way to ensure laser safety in public airspace is to use levels of power density that are inherently eye-safe even in the case of an exposure. For SLR, a major step forward could be to perform ranging at a wavelength near 1.5 µm [31], where the maximum permissible exposure (MPE) is much higher than for the currently used wavelengths of 1064 and 532 nm. For all applications where this approach is not feasible, camera-based aircraft detection will continue to be of high relevance for laser safety. We hope to encourage institutions as well as companies to openly share information on experience with systems for aircraft detection to further improve laser safety in airspace.

APPENDIX A

Figures 7 and 8 show the distribution of the capturing times and dates of the infrared images of the analyzed dataset. The images have been taken during day and night and on 15 different days.

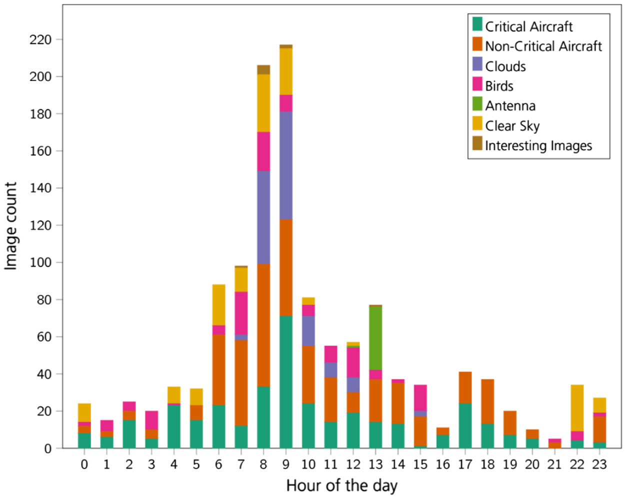

Fig. 7. Distribution of the capturing times of the images in the thermal infrared benchmark over all days of the image campaign [26].

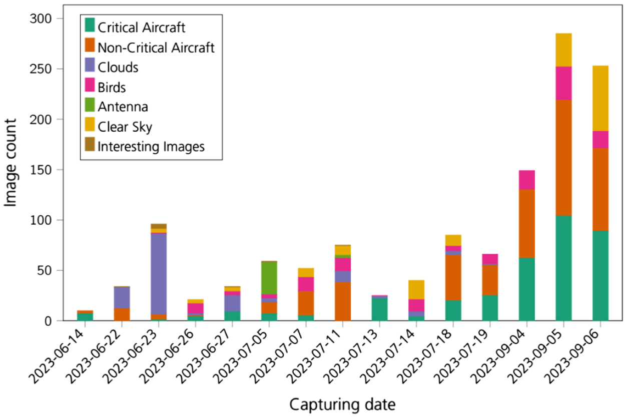

Fig. 8. Distribution of the capturing times of the images in the thermal infrared benchmark over all days of the image campaign [26].

APPENDIX B

The image processing has been tested with Python (version 3.10.11) using the packages “opencv” (version 4.7.0) [32] for image processing, “numpy” (version 1.22.4) [33] for calculations, and “astropy” (version 5.2.2) [34] for extracting images from the “Flexible Image Transport System, FITS” files. The source code (filename “Algorithm.py”) is also provided in a permanent online repository (see data availability statement).

Initially, we import the required packages and methods. The “perf_counter” can be used to measure the execution time of the algorithm.

```
import cv2 as cv
import numpy as np
from astropy.io import fits
from time import perf_counter
```

Subsequently, we define a class “ImageAlgorithm” used for the classification. The steps (see the comments in the source code) directly correspond to those described in Fig. 3.

```
class ImageAlgorithm:
    """Class for performing image analysis using a series of steps."""

    def __init__(self, median_filter_size=5, nr_of_std=12):
        """
        Initialize ImageAlgorithm object.

        Parameters:
            median_filter_size (int): Size of the median filter. Needs to be 3
                                                    or 5.
            nr_of_std (int): Number of standard deviations for thresholding.
        """
        self.median_filter_size = int(median_filter_size)
        self.nr_of_std = int(nr_of_std)

    def analyse(self, image):
        """
        Analyze the input image using a series of steps.

        Parameters:
            image (numpy.ndarray): Input image data.

        Returns:
            dict: A dictionary containing analysis results, including detection
                                                     status,
                  classification, and the resulting image.
        """
        start_time = perf_counter()

        # Step 1: Median filtering
        background = cv.medianBlur(image, self.median_filter_size)

        # Step 2: Image subtraction
        img_without_background = image.astype(int) - background.astype(int)

        # Step 3: Binarization
        binary_image = self._std_thresh(img_without_background, self.nr_of_std)

        # Step 4: Hit-or-miss transform
        final_image = self._hit_miss_operation(binary_image)

        # Step 5: Classification
        detection = bool(np.any(final_image))
        classification = "Unsafe" if detection else "Safe"

        return_dict = {
            ’DETECTED’: detection,
            ’CLASSIFICATION’: classification,
            ’IMAGE’: final_image
        }

        end_time = perf_counter()
        print(f’Image analyzed in {end_time - start_time:.6f}s’)
        return return_dict

    def _std_thresh(self, image, nr_of_std):
        """
        Thresholding based on standard deviation.

        Parameters:
            image (numpy.ndarray): Input image data.
            nr_of_std (int): Number of standard deviations.

        Returns:
            numpy.ndarray: Thresholded image.
        """
        mean = np.mean(image)
        std = np.std(image)
        thresh = mean + nr_of_std * std
        return (np.where(image > thresh, 255, 0)).astype(np.uint8)

    def _hit_miss_operation(self, binary_image):
        """
        Perform hit-or-miss transform.

        Parameters:
            binary_image (numpy.ndarray): Binary image data.

        Returns:
            numpy.ndarray: Resulting image after hit-or-miss operation.
        """
        kernel = np.array([[-1, -1, -1],
                           [-1,  1, -1],
                           [-1, -1, -1]], dtype="int")
        single_pixels = cv.morphologyEx(binary_image, cv.MORPH_HITMISS, kernel)
        single_pixels_inv = np.bitwise_not(single_pixels)
        result = cv.bitwise_and(binary_image, binary_image, mask=
                                                single_pixels_inv)
        return result
```

Finally, we show how the “ImageAlgorithm” class can be used to classify an image. In the code, we initially generate an object called “algorithm” of this class. Subsequently, we import an example image with the filename “Example_Aircraft_Image.fit” and pass it to the “analyse” function of the “algorithm”. This will return a Python dictionary called “results” and we then print the value of the key “Classfication”, which will be a string (either “safe” or “unsafe”), depending on the result of the image analysis.

```
if __name__ == "__main__":
    # Generate an ImageAlgorithm object
    algorithm = ImageAlgorithm()

    # Import an image
    with fits.open(’Example_Aircraft_Image.fit’) as hdul:
        image = hdul[0].data

    # Analyze the image
    result = algorithm.analyse(image)
    print("Result of analysis: " + result[’CLASSIFICATION’])
```

Note that this last code snippet is only used for the purpose of demonstration. In the actual usage of the “ImageAlgorithm” class for real-time aircraft detection, images will not be loaded from the hard drive, but are instead directly processed (in a separate thread), once they are obtained from the camera.

Acknowledgment. The authors would like to thank their colleagues from the department for “administration and support” (within DLR’s Institute of Technical Physics) for their continuing support on improving electronics, mechanics, and documentation of the miniSLR and its components. Furthermore, the authors acknowledge the ILRS for helpful discussions on laser safety and other SLR related topics.

Disclosures. The authors declare no conflicts of interest.

Data availability. Data (thermal infrared images grouped by the different categories) underlying the results presented in this paper as well as the source code for the image processing algorithm are available in Ref. [35].

REFERENCES

1. M. Wilkinson, K. Schreiber, I. Prochazka, *et al.*, "The next generation of satellite laser ranging systems," J. Geod. **93**, 2227–2247 (2018).
2. M. Pearlman, C. Noll, E. Pavlis, *et al.*, "The ILRS: approaching 20 years and planning for the future," J. Geod. **93**, 2161–2180 (2019).
3. M. Steindorfer, G. Kirchner, F. Koidl, *et al.*, "Daylight space debris laser ranging," Nat. Commun. **11**, 3735 (2020).
4. M. T. Knopp, A. Spoerl, M. Gnat, *et al.*, "Towards the utilization of optical ground-to-space links for low earth orbiting spacecraft," Acta Astronaut. **166**, 147–155 (2020).
5. M. Toyoshima, Y. Takayama, T. Takahashi, *et al.*, "Ground-to-satellite laser communication experiments," IEEE Aerosp. Electron. Syst. Mag. **23**(8), 10–18 (2008).
6. S.-K. Liao, W.-Q. Cai, J. Handsteiner, *et al.*, "Satellite-relayed intercontinental quantum network," Phys. Rev. Lett. **120**, 030501 (2018).
7. J. J. Degnan, "Asynchronous laser transponders: a new tool for improved fundamental physics experiments," Int. J. Mod. Phys. D **16**, 2137–2150 (2007).
8. W. Meng, H. Zhang, P. Huang, *et al.*, "Design and experiment of onboard laser time transfer in Chinese Beidou navigation satellites," Adv. Space Res. **51**, 951–958 (2013).
9. E. Cordelli, A. D. Mira, T. Flohrer, *et al.*, "Ground-based laser momentum transfer concept for debris collision avoidance," J. Space Saf. Eng. **9**, 612–624 (2022).
10. S. Scharring and J. Kästel, "Can the orbital debris disease be cured using lasers?" Aerospace **10**, 633 (2023).
11. T. J. Colvin, J. Karcz, and G. Wusk, "Cost and benefit analysis of orbital debris remediation," Tech. Rep. (NASA Headquaters, Office of Technology, Policy, An Strategy, 2023).
12. D. Keil, S. Scharring, E. Klein, *et al.*, "Modification of space debris trajectories through lasers: Dependence of thermal and impulse coupling on material and surface properties," Aerospace **10**, 947 (2023).
13. European organisation for the Safety of Air Navigation, "SRC Doc 7, Outdoor laser operations in the navigable airspace, edition 1," Tech. Rep. (Eurocontrol, 2001).
14. J. McGarry, "NGSLR safety handbook," Tech. Rep. NASA/GSFC/694 (2015).
15. J. Sun, *The 1090 Megahertz Riddle: A Guide to Decoding Mode S and ADS-B Signals*, 2nd ed. (TU Delft OPEN Publishing, 2021).
16. N. Xu, R. Cassell, C. Evers, *et al.*, "Performance assessment of multilateration systems—a solution to nextgen surveillance," in *Integrated Communications, Navigation, and Surveillance Conference Proceedings* (2010).
17. J. T. W. Murphy, "A transponder-based aircraft detector for SLR," *International Workshop on Laser Ranging*, Annapolis, Maryland, 2014.
18. K. E. Wilson, W. Roberts, V. Garkanian, *et al.*, "Plan for safe laser beam propagation from the optical communications telescope laboratory," Progress Report (JPL, Telecommunications and Data Acquisition, 2003), pp 142–152.
19. M. Wilkinson, "SGF, Herstmonceux in-sky safetysystem testing using ADS-B," *ILRS Technical Workshop*, Postdam, Germany, 2016.
20. S. Rudys, P. Ragulis, A. Laučys, *et al.*, "Investigation of uav detection by different solid-state marine radars," Electronics **11**, 2502 (2022).
21. T. Otsubo, M. Kobayashi, Y. Yokota, *et al.*, "Acoustic positioning of closely-flying aircraft for eye safety of space laser applications," Adv. Space Res. **73**, 982–991 (2024).
22. M. Wilkinson, "Optically detecting aircraft for in-sky safety in daylight conditions," *ILRS Technical Workshop*, Stuttgart, Germany, 2019.
23. J. Jakubowicz, S. Lefebvre, F. Maire, *et al.*, "Detecting aircraft with a low-resolution infrared sensor," IEEE Trans. Image Process. **21**, 3034–3041 (2012).
24. A. Leidig, U. Schreiber, T. Bachem, *et al.*, "Free space laser safety system for aircraft camera detection in the infrared," *ILRS Technical Workshop*, Stuttgart, Germany, 2019.
25. L. Li, X. Zhou, Z. Hu, *et al.*, "On-orbit monitoring flying aircraft day and night based on SDGSAT-1 thermal infrared dataset," Remote Sens. Environ. **298**, 113840 (2023).
26. J. Steurer, "Development of a system for airborne object detection to ensure safe operation of laser-optical ground stations," Master's thesis (University of Stuttgart, 2024).
27. D. Hampf, F. Niebler, T. Meyer, *et al.*, "The miniSLR: a low-budget, high-performance satellite laser ranging ground station," J. Geod. **98**, 8 (2024).
28. W. Linder, *Digital Photogrammetry* (Springer Berlin Heidelberg, 2009).
29. S. Perreault and P. Hebert, "Median filtering in constant time," IEEE Trans. Image Process. **16**, 2389–2394 (2007).
30. I. Mednieks, "Object detection in grayscale images based on covariance features," in *International Conference on Signals and Electronic Systems* (IEEE, 2008).
31. H. Kunimori, B. Greene, K. Hamal, *et al.*, "Centimetre precision eye-safe satellite laser ranging using a Raman-shifted Nd:YAG laser and germanium photon counter," J. Opt. A **2**, 1 (2000).
32. G. Bradski, "The OpenCV Library," Dr. Dobb's J. Softw. Tools **120**, 122–125 (2000).
33. C. R. Harris, K. J. Millman, S. J. van der Walt, *et al.*, "Array programming with NumPy," Nature **585**, 357–362 (2020).
34. The Astropy Collaboration, A. M. Price-Whelan and P. L. Lim, "The Astropy project: Sustaining and growing a community-oriented open-source project and the latest major release (v5.0) of the core package," Astrophys. J. **935**, 167 (2022).
35. J. Steurer, N. Bartels, D. Hampf, *et al.*, "Dataset of thermal infrared images," figshare. 2024, https://doi.org/10.6084/m9.figshare.25738860.v1.