sensors

MDPI

Article

ThermoSteg—Covert Channel for Microbolometer Thermographic Cameras

Krzysztof Sawicki *, Grzegorz Bieszczad and Tomasz Sosnowski

Institute of Optoelectronics, Military University of Technology, 00-908 Warsaw, Poland; grzegorz.bieszczad@wat.edu.pl (G.B.); tomasz.sosnowski@wat.edu.pl (T.S.)
* Correspondence: krzysztof.sawicki@wat.edu.pl; Tel.: +48-261-83-94-23

Abstract: The article presents a new concept—steganography in thermography. Steganography is a technique of hiding information in a non-obvious way and belongs to sciences related to information security. The proposed method, called ThermoSteg, uses a modification of one of the parameters of the thermal imaging camera—integration time—to embed the signal containing hidden information. Integration time changing makes the microbolometer array heat up while reading the sensors. The covert information can be extracted from the stream of thermograms recorded by another thermal camera that observes the first one. The covert channel created with the ThermoSteg method allows the transmission of covert data using a thermal sensor as a wireless data transmitter. This article describes a physical phenomenon that is exploited by the ThermoSteg method and two proposed methods of covert data extraction, and presents the results of experiments.

Keywords: steganography; hardware security; thermography; covert channel; thermal cameras

Citation: Sawicki, K.; Bieszczad, G.; Sosnowski, T. ThermoSteg—Covert Channel for Microbolometer Thermographic Cameras. *Sensors* **2021**, *21*, 6395. https://doi.org/10.3390/s21196395

Academic Editor: Thomas Udelhoven

Received: 10 August 2021
Accepted: 21 September 2021
Published: 24 September 2021

1. Introduction

Information is now one of the essential goods, and information security mechanisms are developing very quickly. For most of the usage scenarios, the best way to secure data is to use cryptography. However, in some specific cases, another method can be used—steganography [1–4]. Steganography is a technique that enables data transmission in a hidden way. For steganography, the critical factor is to make the data 'covert', so that no one would be able to receive the information. This approach differs from cryptography's approach, where the data can be received by virtually anyone, but cannot be decrypted by anyone except the authorised recipient.

Steganography creates so-called covert channels—communication channels where the data are transmitted in a hidden manner. The key element of these covert channels is the method of hiding data, which should be kept secret similar to the way in which the encryption key in cryptography is kept secret. Covert channels can be designed with different approaches: using unused bits in network protocol headers, modifying time-dependent parameters, or by using phenomena that are, in most cases, treated as unwanted or random. Many of the covert channels provide a bitrate smaller than tens of bits per second [4]. The very small bitrate makes the imperceptibility of the covert channel better according to the "magic triangle of steganography". The covert channel should be imperceptible, robust and have a large capacity (bitrate). The rule of the "magic triangle" says that only two of these features can be achieved and the third feature will not be fully implemented (Figure 1) [5,6]. There are sophisticated covert channels that provide a bitrate on the order of tenths of a bit per second. The perfect example of such a covert channel is BitWhisper [7], where the authors achieved a bitrate of 8 bits per hour (2.22×10^{-3} bps). Such low values make it impossible to use covert channels for audio or video transmission, but they are sufficient for signalling or the basic control of devices. Covert channels can be used when the fact of the presence of the communication should be kept secret, for example when the sensors that are placed in the enemy's territory or on

commonly used devices have a dual purpose, where secondary functionality is meant to be secret. Steganographic communication between sensors makes it harder to discover them. Steganographic communication might also be used as a part of so-called hardware trojans.

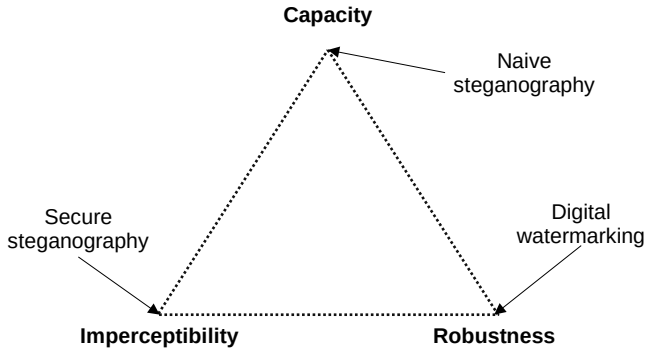

Figure 1. "Magic triangle of steganography." [5,6].

Covert channels and their usage are a popular research topic. The principles of covert channel creation were described in 1989 in Wolf [8]. Since then, many new and more sophisticated types of covert channels have been published. One of the most exploited fields is computer network steganography, where two types of covert channels can be distinguished—timing covert channels and storage covert channels. The first type uses the modulation of the interval between two network events, such as the beginning of packet transmissions in wireless channels [9] or adjusting silence periods in VoLTE transmission [10]. The other type, storage channels, exploits unused or partially used fields in network protocols, such as the Timestamp field in IEEE 802.11 Beacon frames [11]. Covert channels can also be embedded in other types of media, such as audio [12] and video [13] streams. It is also possible to create a storage covert channel in a phase drift of signals modulated with QAM modulations [14,15].

Such a broad selection of different covert channel types makes them an interesting subject for computer security teams. Covert channels can be a potential security breach and can be used for data leakage from network-isolated computer systems via 'Air-Gap'. Many experiments of this type have been performed by Mordechai Guri [16–18] and other researchers [19–21]. For example, it is possible to use very low contrast or fast flickering images, which are invisible to human subjects, to transmit data using a computer display [22]. Another way is to use relatively cheap hardware to detect electromagnetic emissions from a USB [23]. Temperature can also be used as a medium for a covert channel. Two network-isolated computers can communicate via a covert thermal channel by stimulating CPU load on one computer as a transmitter and reading temperature sensors on the other computer as a receiver [7]. Covert channels created this way enable covert transmission with a bitrate of 8 bits per hour at a distance of up to 40 cm. Another example of hiding data in a non-obvious way with the use of thermal signals is presented in [24]. The method presented is based on the active heating of material by means of laser radiation. Unfortunately, the authors do not provide any bandwidth estimates of the proposed method.

This paper proposes a new type of covert channel that utilizes thermal cameras and their sensors to make steganographic communication possible. The covert channel is established between two thermal cameras, where one is acting as a transmitter, and the other can receive data. The steganographic transmission is possible thanks to the modification of some operational parameters of custom made microbolometric thermal cameras created for navigation systems [25]. The proposed covert channel can be classified as a timing channel because the data are hidden in the time characteristics of the thermal signal acquired from the part of the microbolometric sensors matrix that corresponds to the thermal image of the other's camera sensor. The method uses a non-obvious way of

modifying the camera's parameters to enable transmission capabilities in microbolometric detectors.

2. Principles of the Method

The covert channel is a particular example of a telecommunication channel. In every communication channel, there must be a transmitter and a receiver. One camera acts as a hidden information transmitter (Tx) and the other as a receiver (Rx). Both cameras' primary function is that of an ordinary thermal imaging camera. Such cameras could be an element of a security system or smart building infrastructure. In the proposed solution, two thermal imaging cameras are facing each other, as shown in Figure 2.

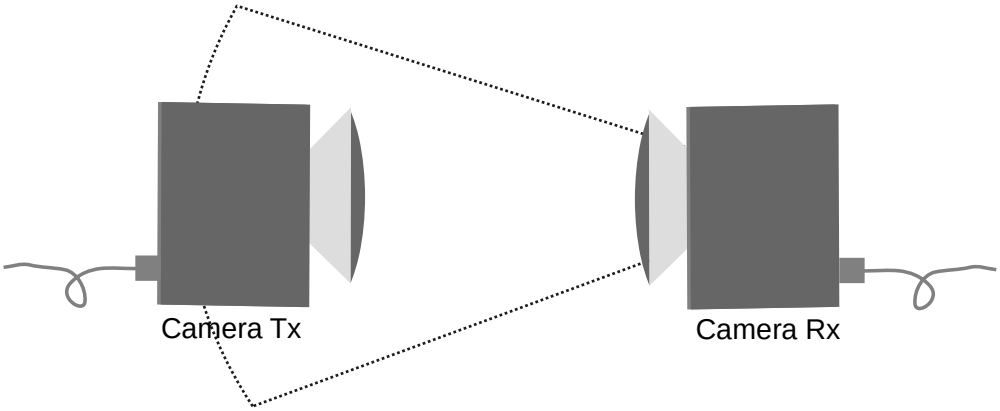

Figure 2. The concept of the covert channel using two thermal cameras, one for data transmission (Tx) and one for data reception (Rx).

The standard reading procedure needed to produce the thermal image is to retrieve data from the successive rows of thermal sensors. When the row of sensors in the matrix is read, an electric current flows through it. This current causes a temporary increase in the microbolometers' temperature, according to Joule's Law. This phenomenon in the microbolometer arrays is called self-heating [26]. Such a phenomenon is shown on a thermal image made with an FLIR SC7900VL and with a microscopic lens, presented in Figure 3.

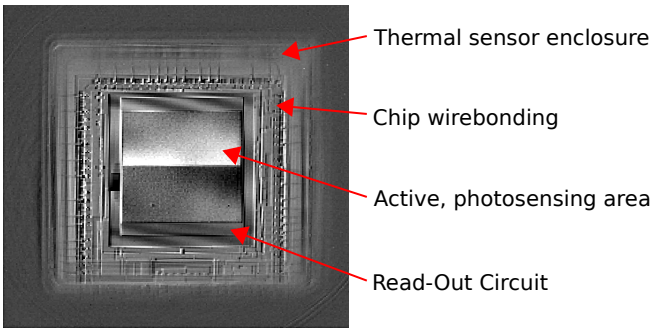

Figure 3. Thermal image of Micro80 detector array, made with a microscopic thermal camera.

The temperature change is significant enough to be sensed by a remote sensor, and for the exemplary case shown in the picture above, it reaches 2.2 °C in temperature amplitude. The secondary remote sensor, sensing the self-heating phenomenon, can be, for example, another thermal camera. This secondary thermal camera (Rx) facing the camera with a microbolometric sensor (Tx) can provide a thermal signal visible in the thermogram.

In the area observed by the Rx camera, it is possible to distinguish the area (ROI—Region of Interest) in which the Tx camera lens is visible. Thanks to the fact that the thermal camera lens is transparent to the infrared spectrum, the thermal signal produced in the Tx camera by the self-heating phenomenon can be sensed remotely through the lens.

This signal can be sensed by the Rx camera by observing the area that the Tx camera is occupying. The area is dependent on the Tx camera's lens size, the Rx camera's focal length and the distance between the cameras. An exemplary image of such an observed lens has a size of $X_p \cdot Y_p$ pixels and is marked in Figure 4, presenting the image obtained by the Rx Camera observing the Tx Camera. It should be noted that both cameras operate normally, which enables the registration of regular thermograms in both cameras.

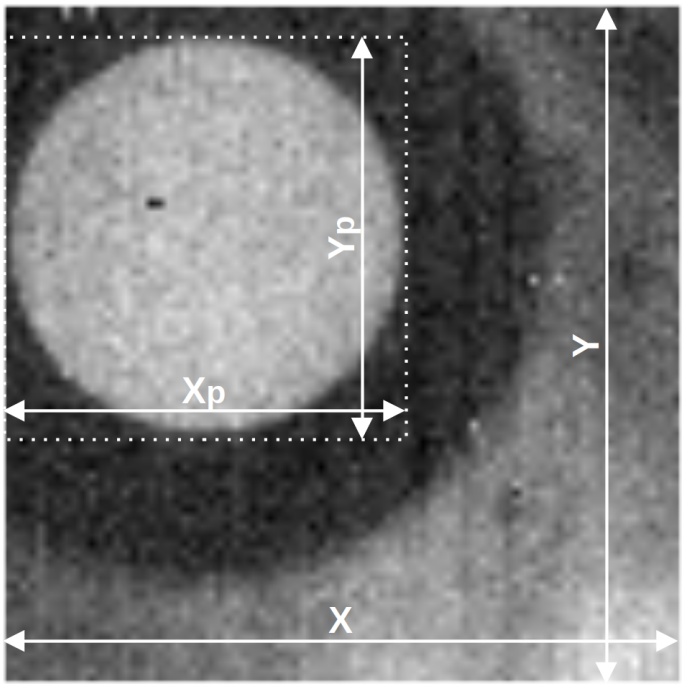

Figure 4. Image captured by Rx camera after nonuniformity correction. In the highlighted square, the Tx camera's lens is visible.

2.1. Covert Data Coding and Embedding

Coding of the covert data is performed by changing one of the main operational parameters of the microbolometer focal plane array—the integration time (t_i). Integration time regulates how long the measurement current from the Readout circuit flows through the row of bolometers in the array, which makes the self-heating phenomenon controllable. Reducing this time causes the current to flow shorter through the row in the bolometer, which results in less heating of the elements of this row; similarly, increasing t_i will cause the elements of the currently read line to heat up more. These temperature differences can be detected with the use of the receiving camera. A detailed description of the operation of the readout circuit in the microbolometer array and its thermodynamic consideration are presented in [26]. The covert data are bivalently encoded using two different integration times t_{i1} and t_{i2}. The hidden data embedding process is shown in Figure 5—the digital signal U_{int}, with the pulse width encoded by the covert data, controls the length of the integration process in the integration circuit. This paper considers the case when the covert data are binary. For this reason, the set of different integration times contains two values. It is possible to extend the method to use three or more different integration times to encode more values at a time.

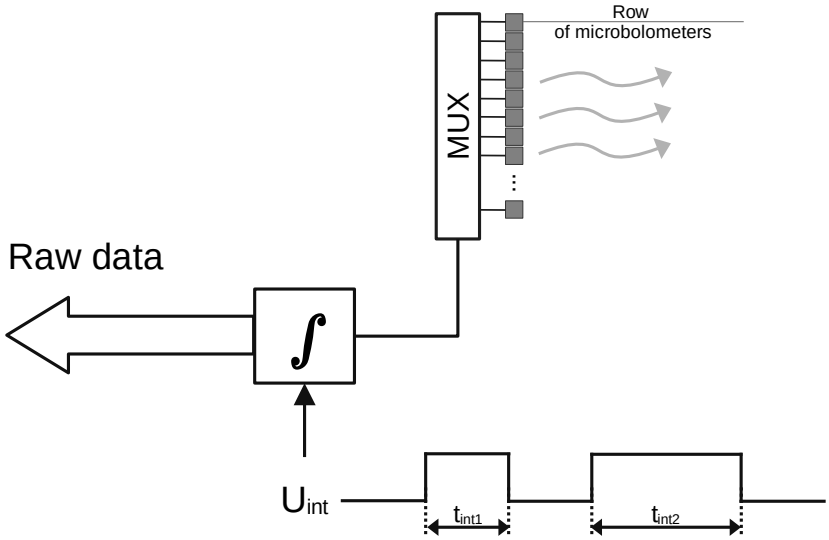

Figure 5. Data embedding scheme.

2.2. Covert Data Reception—Amplitude Analysis

The ROI observation area contains $X_p \cdot Y_p$ pixels. Each observation is a subject of noise that is dependent on the NETD of the thermal camera and the optical path attenuation. The signal from the single detector can have an insufficient signal-to-noise ratio to extract subtle temperature changes in the observed microbolometric camera. To increase the signal-to-noise ratio, the spatial averaging is conducted in such a way that the average value of all F_n pixels observed in the ROI is computed for each thermogram according to the formula:

$$F(n) = \frac{\sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} p_{xy}}{(x_2 - x_1)(y_2 - y_1)}, \quad (1)$$

where p_{xy} is the value of the pixel with the coordinates (x,y) in the n-th thermogram, x_1 and x_2 are the numbers of the first and last columns of the analyzed area, while y_1 and y_2 are the numbers of the first and the last lines of the analyzed area.

This creates the signal $F = \{F_0, F_1, ..., F_N\}$ of values from the averaged pictures, which constitutes the ROI's temperature signal sampled with the camera operating frequency f_p. An example of the F signal is shown in Figure 6.

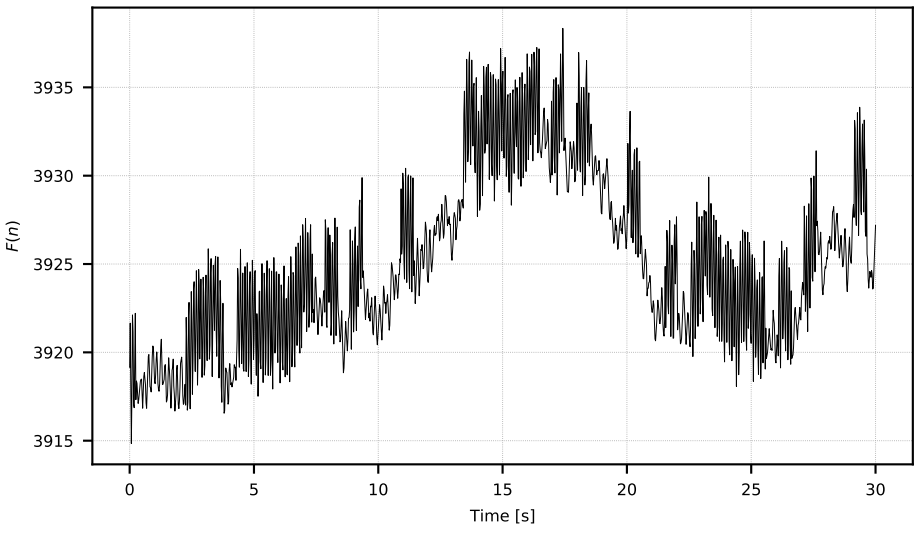

Figure 6. Exemplary F signal exhibiting strong 1/f noise.

The F signal shows noticeable low-frequency noise resulting from the influence of external factors, that is, changes in the temperature of the camera's surroundings. This noise has a typical 1/f spectral density and is very common in thermal sensors [27–29]. The spectral noise character and usable signal encoding scheme make it possible to separate one from another by means of a temporal filter. For further analysis, only the changes of higher frequency that interest us should be extracted; this is performed with the filter described by the equation:

$$F'(n) = F(n) - \frac{\sum_{i=n-w}^{n} F(i)}{w} \; for \; n = \{w, w+1, ..., N\}, \tag{2}$$

where $F(n)$ is the n-th sample of the F signal, $F'(n)$ is the n-th sample of the F' signal, N is the length of the F signal, w is the width of the applied window.

The width of the window w should be selected experimentally according to the spectral characteristic of the low frequency noise. The exemplary resulting F' signal is shown in Figure 7.

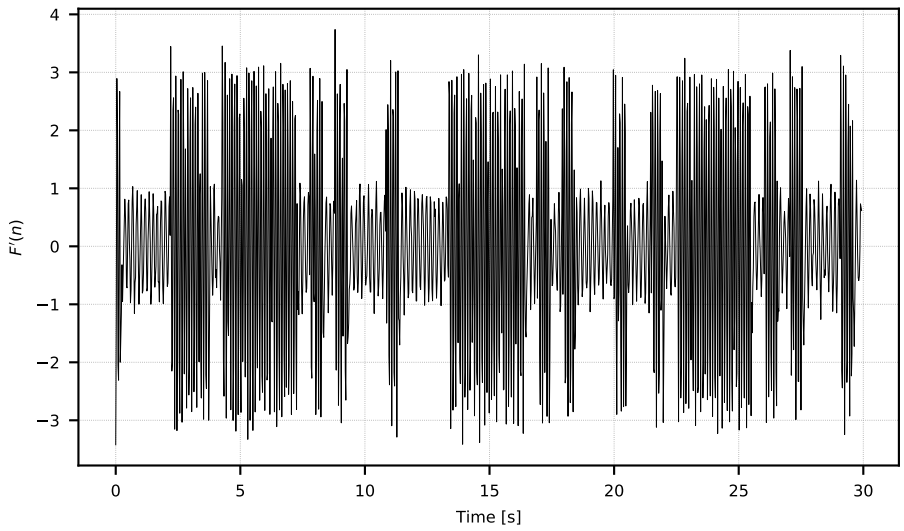

Figure 7. Signal F' after high-pass filtering.

In Figure 7, one can see the moments of the higher and lower amplitudes of the signal. The amplitude changes are the direct consequences of the integration time manipulation in the observed microbolometric array. Changes in signal amplitude can be easily estimated with power metrics according to the Formula (3). The resulting signal used for the analysis is shown in Figure 8.

$$F'_p(n) = [F'(n)]^2. \tag{3}$$

The F'_p signal still exhibits some noise, which is why it is then averaged over a temporal window of length w'. The result of this averaging is the F'_f signal calculated with (4) and presented in Figure 9:

$$F'_f(n) = \frac{\sum_{i=n-w'}^{n} F'_p(i)}{w'}. \tag{4}$$

The value of w' should be selected experimentally according to the characteristics of the signal received.

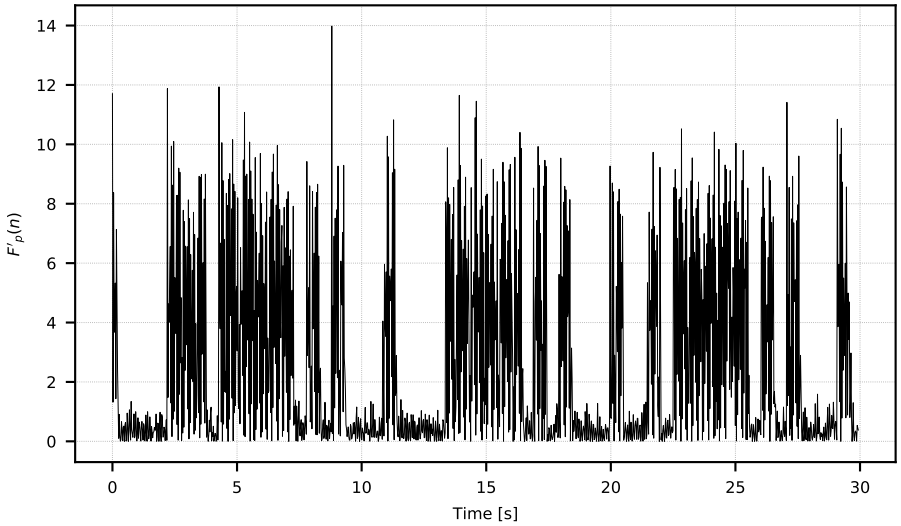

Figure 8. Signal F'_p, which is the power of the filtered signal F'.

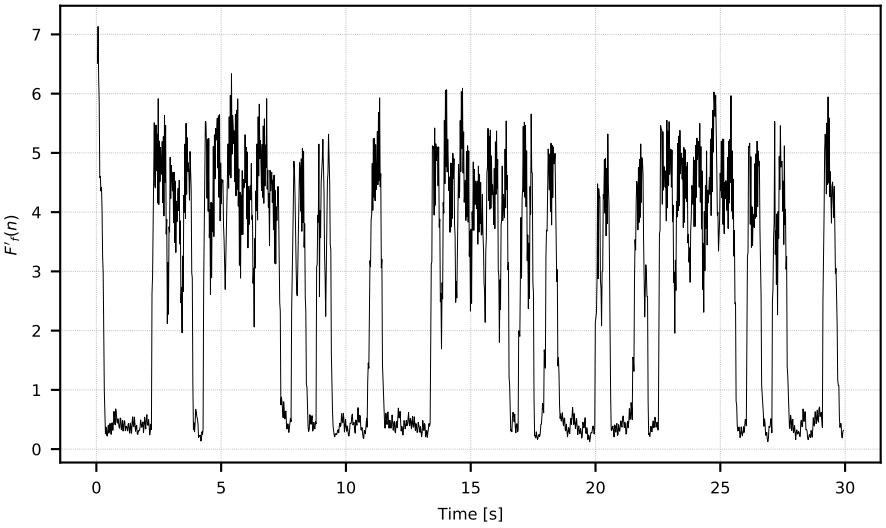

Figure 9. Signal F'_f averaged with window size $w' = 5$.

In signal F'_f, one can clearly see the characteristics of the binary waveform. To create a binary $F'b$ signal, it is required to perform a threshold classification of the samples:

$$F'_b(n) = \begin{cases} 1 \; when \; F'_f(n) \geq \overline{F'_f} \\ 0 \; when \; F'_f(n) < \overline{F'_f}, \end{cases} \tag{5}$$

where $\overline{F'_f}$ is the average value of all samples of the F'_f signal. An example of the F'_b signal is shown in Figure 10. The signal character corresponds to the signal used to supply the integration time changes in the Tx camera. The F'_b binary signal proves that hidden information is embedded in the F received signal.

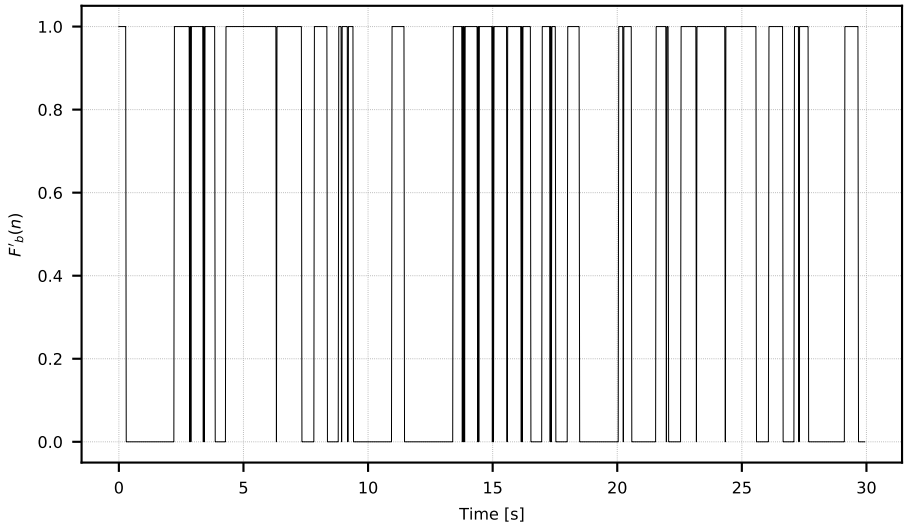

Figure 10. Received binary signal F'_b.

2.3. Covert Data Reception—Variance Analysis

The algorithm's effectiveness depends on the thermal amplitude of the received signal that is the consequence of the integration time values t_i, chosen to encode the thermal signal in the Tx camera. For a low signal amplitude caused, for example, by a large distance between the receiving and transmitting agents or lower lens transmission, amplitude demodulation can cause a high error rate. For such a situation, detection based on the signal variance analysis has been developed.

Having the $F = \{F_0, F_1, ..., F_n\}$ signal calculated on the basis of (1), one can calculate the value of the F'' signal consisting of the value of the standard deviation of the F signal calculated in a temporal window with a width of w_s according to Equation (6). Signal $F''(n)$ is also presented in Figure 11.

$$F''(n) = \sqrt{\frac{1}{w_s} \sum_{i=n-w_s}^{n} [F(i) - (\frac{1}{w_s} \sum_{j=n-w_s}^{n} F(j))]^2}. \quad (6)$$

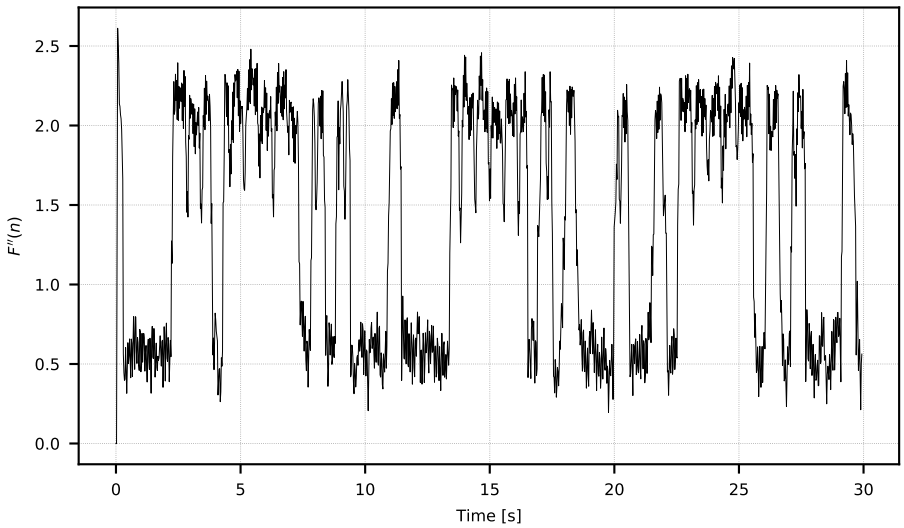

Figure 11. Signal F'' (6) calculated with $w_s = 5$.

In Equation (6), w_s denotes the width of the window in which the standard deviation is calculated and $F''(n)$ denotes the n-th sample of the signal F''. The F'' signal should be

filtered (7) to obtain the F''_f signal (Figure 12). Then, (8) should be classified, as a result of which the signal F''_b will be obtained.

$$F''_f(n) = \frac{\sum_{i=n-w''}^{n} F''(i)}{w''} \tag{7}$$

$$F''_b(n) = \begin{cases} 1 \; when \; F''_f(n) \geq \overline{F''_f} \\ 0 \; when \; F''_f(n) < \overline{F''_f}. \end{cases} \tag{8}$$

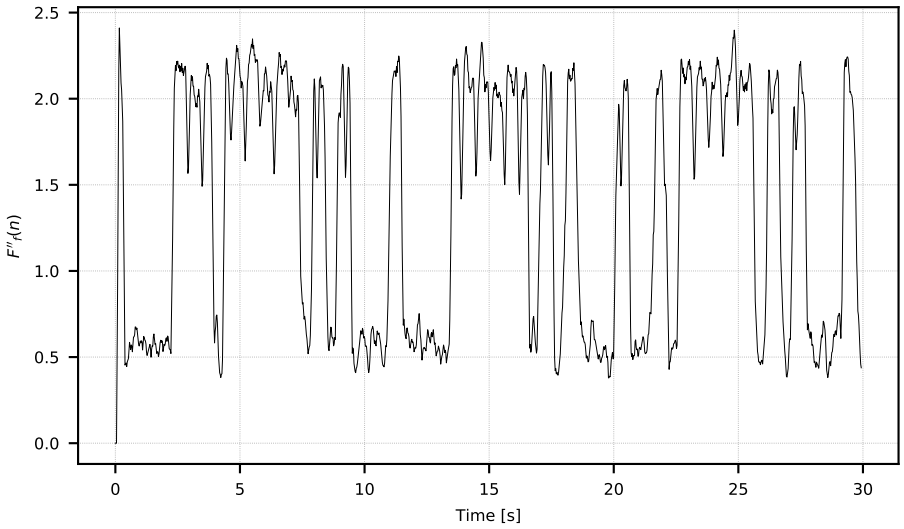

Figure 12. Signal F''_f calculated with $w'' = 4$.

2.4. Covert Data Extraction

The sampling frequency of F'_b and F''_b binary signals is equal to the frequency of camera operation f_p. The covert data stream is sampled with a lower frequency, and every covert bit is conveyed by the specific number of equal value samples of F'_b and F''_b signals. This number is defined as B:

$$B = \frac{f_p}{W}, \tag{9}$$

where W is the assumed hidden binary bit rate (e.g., 2 bps).

Based on this information, the binary sequence decoding process is performed. The algorithm written in Python finds the first change in the value of the sample input signal F' or F'' considering it the beginning of the covert bit. Then the algorithm averages the value of B samples counted from the beginning of the covert bit. If the mean value of these samples is greater than 0.5 then this is classified as the 1 covert bit. Otherwise, it is the 0 covert bit. Because some samples in the signals F' and F'' may be lost, the algorithm is able to synchronize to the binary string. The decoding algorithm is presented in Algorithm 1.

Algorithm 1 Decoding algorithm.

```
procedure DECODESIGNAL(data, B, correction)
    decodedData ← [ ]                                   ▷ Initialize an empty vector
    i ← 1
    bl ← 0
    recvBit ← 0
    while data[i] = data[0] do
        i ← i + 1
    end while
    repeat
        if i + b > length(data) then
            break
        end if
        if average(data[i : i + B]) > 0.5 then
            recvBit ← 1
        else
            recvBit ← 0
        end if
        append(recvBit, decodedData)          ▷ Append recvBit to the end of the vector
        i ← i + B
        if correction = 1 and i < length(data) and data[i] ≠ recvBit then
            offset ← correction
            while offset > 0 do
                if data[i] = data[i − offset] then
                    i ← i − offset
                    break
                end if
                offset ← offset − 1
            end while
        end if
    until break
    return decodedData
end procedure
```

3. Experiments

3.1. Equipment

Two thermal imaging cameras based on FPGA Cyclone V and Lynred Micro80 matrices (80 × 80 pixels) [25] were used to perform the experiments. The camera matrices were positioned opposite each other at a distance of 14 cm, as shown in Figure 13. The cameras worked under the control of the GNU/Linux system and transmitted data using a dedicated protocol based on UDP over a network operating in the Gigabit Ethernet standard. The cameras generated thermograms with the frequency $f_p = 44$ Hz, as it can be easily divided by two and by four. Further analysis was carried out using software written in Python on a PC.

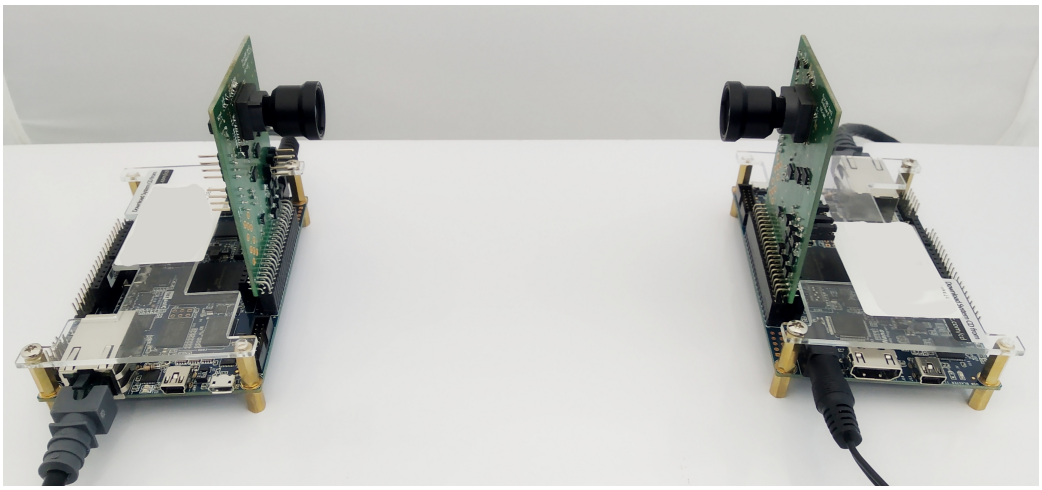

Figure 13. Camera setup for the method evaluation.

3.2. Results and Discussion

In order to determine the parameters of the method, attempts were made to transmit 1024 data packets of 10 bits each (every possible combination). Data packets were preceded by a one-bit preamble, which served as the start of the packet mark. The end of the data packet was also signalled with one stop-bit. The experiment was repeated for four different assumed binary bit rates $W = \{0.5,\ 1,\ 2,\ 4\}$ b/s and three sets of integration times $t_{i1} = (82.05\ \mu s; 328.21\ \mu s)$, $t_{i2} = (123.08\ \mu s; 287.18\ \mu s)$, $t_{i3} = (164.10\ \mu s; 246.15\ \mu s)$. The bit error rate was determined using the amplitude analysis method and the amplitude variability analysis method for each set of integration times and binary bit rates. Any errors in the preambles were not taken into account when determining the bit error rate. Computed BER values are presented in Tables 1 and 2. Graphical comparisons of the results are shown in Figures 14–16.

Table 1. Bit error rate determined for the amplitude analysis method.

t_{int} / W	t_{i1}	t_{i2}	t_{i3}
0.5 b/s	2.12%	2.55%	7.69%
1 b/s	2.97%	4.07%	9.87%
2 b/s	6.45%	6.50%	12.67%
4 b/s	13.00%	31.62%	35.97%

Table 2. Bit error rate determined for the amplitude variance analysis method.

t_{int} / W	t_{i1}	t_{i2}	t_{i3}
0.5 b/s	1.69%	1.20%	10.00%
1 b/s	2.04%	2.41%	12.12%
2 b/s	5.57%	4.49%	25.47%
4 b/s	13.67%	12.46%	36.58%

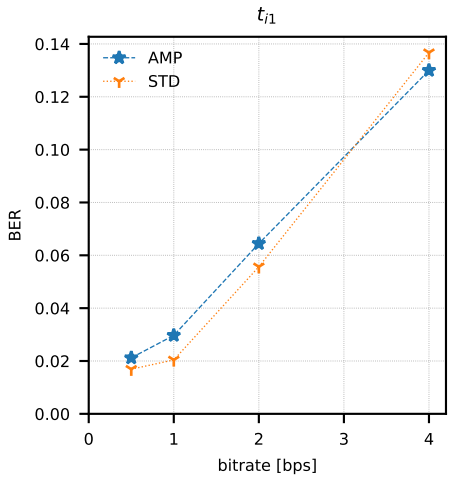

Figure 14. Bit error rate values as a function of bit rate and analysis method (AMP—amplitude analysis, STD—variance analysis) for t_{i1} integration time set.

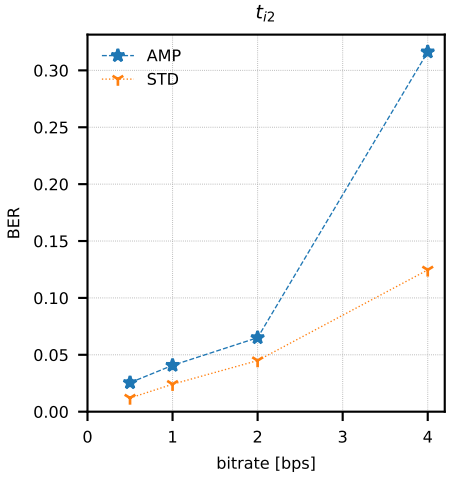

Figure 15. Bit error rate values as a function of bit rate and analysis method (AMP—amplitude analysis, STD—variance analysis) for t_{i2} integration time set.

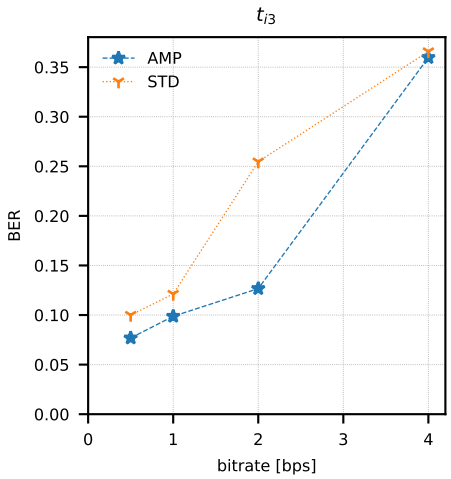

Figure 16. Bit error rate values as a function of bit rate and analysis method (AMP—amplitude analysis, STD—variance analysis) for t_{i3} integration time set.

Both methods of analysis produced similar results for the recorded transmissions. For t_{i1} and t_{i2}, the variable analysis looks to be more promising as the BER values are almost

two times better. The results create a chance for error-free transmission with the use of correction and detection codes in the information layer.

The analysis of error distribution in received messages was also performed. This analysis also shows how many messages have been received without errors. It is easy to observe that this distribution has the character of a Poisson distribution with an expected value equal to 0 in most cases. A different expected value was observed only with the highest bit rates, mostly with t_{i3} integration times set. Results of the analysis are presented in Figures 17–19. These figures show that most of the received messages were decoded with at most one error bit, which can be easily corrected with simple correction methods. They also show a slightly better number of correct messages decoded with amplitude variance analysis for t_{i1} and t_{i2} integration time sets.

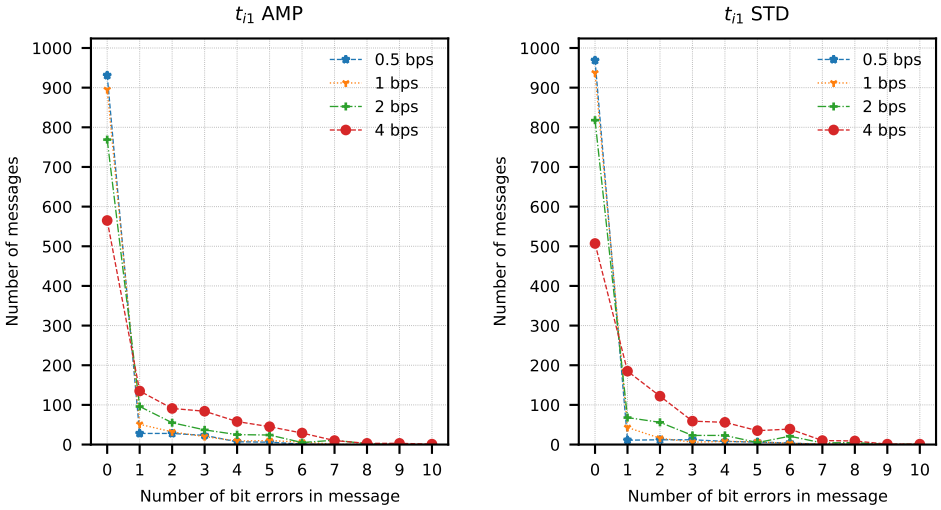

Figure 17. Number of bit errors in messages transmitted with t_{i1} set calculated with amplitude analysis method (AMP) and amplitude variance analysis method (STD).

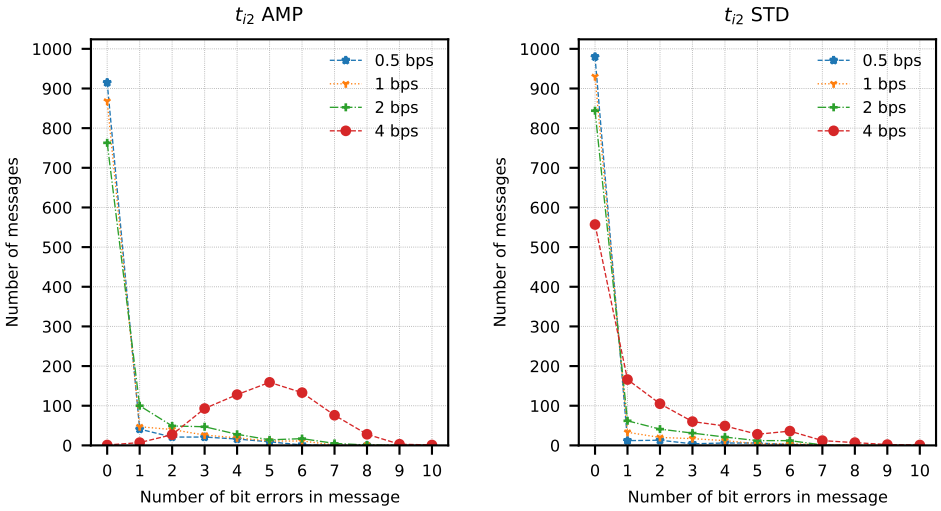

Figure 18. Number of bit errors in messages transmitted with t_{i2} set calculated with amplitude analysis method (AMP) and amplitude variance analysis method (STD).

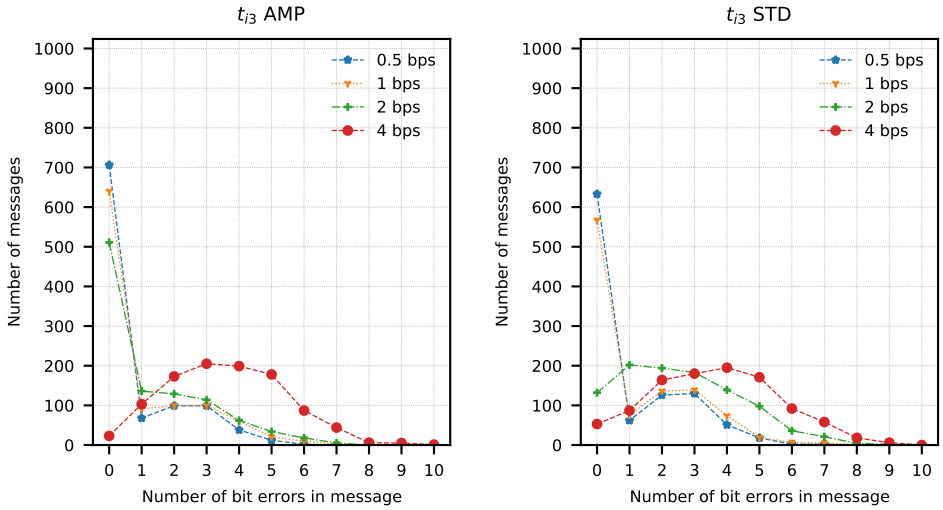

Figure 19. Number of bit errors in messages transmitted with t_{i3} set calculated with amplitude analysis method (AMP) and amplitude variance analysis method (STD).

Broadcasting information hidden by a thermal imaging camera does not disturb its primary task of producing a usable thermal image. In such a case, however, one should be aware of the need to introduce suitable correction of the thermograms, taking into account the changing values of the integration time. The difference between thermograms acquired with two different integration times is shown in Figure 20. With a shorter integration time, the dynamics of the output raw signal from the sensors are lower. Some degradation of thermal resolution (NETD) of the thermal camera will be present due to integration time manipulation. Because the change of the integration time still produces usable thermograms, there is a theoretical possibility of creating a full-duplex link with this method.

t_{int1} t_{int2}

Figure 20. Two thermograms of the same scene acquired with different integration times ($t_{int1} = 82.05$ µs; $t_{int2} = 328.21$ µs).

4. Conclusions

The hidden transmission methods do not ensure high bit rates. Hence, their scope of use is limited, for example, to the transmission of encryption keys or emergency device control commands. The proposed method can be used to create a communication channel between thermal imaging cameras transmitting, for example, encryption keys, performing camera authentication or the detection of unauthorized devices to prevent counterfeits or to eliminate rogue devices. This method can also be used to create a diagnostic interface with the thermal cameras that are mounted in inaccessible places such as a fire control system camera in a tank, or to transmit, for example, the coordinates of the transmitting camera, thanks to which the recorded thermograms can be supplemented with the information about the parameters of the monitored area. For such applications, bit rates of the order of

single bits per second are sufficient. The undoubted advantage of the proposed method is its undetectability with the use of radio communication analysis equipment.

The proposed method has a limited operating range but can be used in dense networks of thermal imaging sensors, where the distances between the cameras are relatively small. Combined with the methods of infrared camera detection, this can increase the security of such networks.

5. Patents

The ThermoSteg method is a patent pending with application No. WIPO ST 10/C PL437673.

Author Contributions: Conceptualization, K.S. and G.B.; methodology, K.S.; software, K.S.; validation, K.S and T.S.; formal analysis, T.S.; resources, G.B.; writing—original draft preparation, K.S.; writing—review and editing, G.B. and T.S.; funding acquisition, G.B. All authors have read and agreed to the published version of the manuscript.

Funding: The project was made using hardware produced in the project financed by the National Centre for Research and Development with project No DOB-2P/02/09/2018. and partly from research funds of the Institute of Optoelectronics; Military University of Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: No data available.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Cox, I.; Miller, M.; Bloom, J.; Fridrich, J.; Kalker, T. *Digital Watermarking and Steganography*; Morgan Kaufmann: Burlington, MA, USA, 2007.
2. Shih, F.Y. *Digital Watermarking and Steganography: Fundamentals and Techniques*; CRC Press: Boca Raton, FL, USA, 2017.
3. Lubacz, J.; Mazurczyk, W.; Szczypiorski, K. Principles and overview of network steganography. *IEEE Commun. Mag.* **2014**, *52*, 225–229. [CrossRef]
4. Mishra, R.; Bhanodiya, P. A review on steganography and cryptography. In Proceedings of the 2015 International Conference on Advances in Computer Engineering and Applications, Ghaziabad, India, 19–20 March 2015; pp. 119–122.
5. Grībermans, D.; Jeršovs, A.; Rusakovs, P. Development of requirements specification for steganographic systems. *Appl. Comput. Syst.* **2016**, *20*, 40–48. [CrossRef]
6. Hamid, N.; Yahya, A.; Ahmad, R.B.; Al-Qershi, O.M. Image steganography techniques: An overview. *Int. J. Comput. Sci. Secur.* **2012**, *6*, 168–187.
7. Guri, M.; Monitz, M.; Mirski, Y.; Elovici, Y. Bitwhisper: Covert signaling channel between air-gapped computers using thermal manipulations. In Proceedings of the 2015 IEEE 28th Computer Security Foundations Symposium, Verona, Italy, 13–17 July 2015; pp. 276–289. [CrossRef]
8. Wolf, M. Covert channels in LAN protocols. In *Local Area Network Security Workshop*; Springer: Berlin, Germany, 1989; pp. 89–101. [CrossRef]
9. Tahmasbi, F.; Moghim, N.; Mahdavi, M. Adaptive ternary timing covert channel in IEEE 802.11. *Secur. Commun. Netw.* **2016**, *9*, 3388–3400. [CrossRef]
10. Zhang, X.; Tan, Y.A.; Liang, C.; Li, Y.; Li, J. A covert channel over volte via adjusting silence periods. *IEEE Access* **2018**, *6*, 9292–9302. [CrossRef]
11. Sawicki, K.; Piotrowski, Z. The proposal of IEEE 802.11 network access point authentication mechanism using a covert channel. In Proceedings of the 2012 19th International Conference on Microwaves, Radar & Wireless Communications, Warsaw, Poland, 21–23 May 2012; Volume 2, pp. 656–659. [CrossRef]
12. Tabara, B.; Wojtuń, J.; Piotrowski, Z. Data hiding method in speech using echo embedding and voicing correction. In Proceedings of the 2017 Signal Processing Symposium (SPSympo), Jachranka, Poland, 12–14 September 2017; pp. 1–6. [CrossRef]
13. Lenarczyk, P.; Piotrowski, Z. Novel hybrid blind digital image watermarking in Cepstrum and DCT domain. In Proceedings of the 2010 International Conference on Multimedia Information Networking and Security, Nanjing, China, 4–6 November 2010; pp. 356–361. [CrossRef]
14. Grzesiak, K.; Piotrowski, Z.; Kelner, J.M. A Wireless Covert Channel Based on Dirty Constellation with Phase Drift. *Electronics* **2021**, *10*, 647. [CrossRef]

15. Cao, P.; Liu, W.; Liu, G.; Ji, X.; Zhai, J.; Dai, Y. A wireless covert channel based on constellation shaping modulation. *Secur. Commun. Netw.* **2018**, *2018*, 1214681. [CrossRef]
16. Guri, M.; Solewicz, Y.; Elovici, Y. Speaker-to-speaker covert ultrasonic communication. *J. Inf. Secur. Appl.* **2020**, *51*, 102458. [CrossRef]
17. Guri, M. Magneto: Covert channel between air-gapped systems and nearby smartphones via cpu-generated magnetic fields. *Future Gener. Comput. Syst.* **2021**, *115*, 115–125. [CrossRef]
18. Guri, M. Exfiltrating data from air-gapped computers via ViBrAtIoNs. *Future Gener. Comput. Syst.* **2021**, *122*, 69–81. [CrossRef]
19. Zhan, Z.; Zhang, Z.; Koutsoukos, X. Bitjabber: The world's fastest electromagnetic covert channel. In Proceedings of the 2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST), San Jose, CA, USA, 7–11 December 2020; pp. 35–45. [CrossRef]
20. Zhang, J.; Ji, X.; Xu, W.; Chen, Y.C.; Tang, Y.; Qu, G. MagView: A Distributed Magnetic Covert Channel via Video Encoding and Decoding. In Proceedings of the IEEE INFOCOM 2020-IEEE Conference on Computer Communications, Toronto, ON, Canada, 6–9 July 2020; pp. 357–366. [CrossRef]
21. Li, L.; Lu, Y.; Yan, X.; Tan, D. Exfiltrating data from an air-gapped system through a screen-camera covert channel. *Math. Biosci. Eng.* **2019**, *16*, 7458–7476. [CrossRef] [PubMed]
22. Guri, M.; Hasson, O.; Kedma, G.; Elovici, Y. An optical covert-channel to leak data through an air-gap. In Proceedings of the 2016 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 12–14 December 2016; pp. 642–649.
23. Guri, M.; Monitz, M.; Elovici, Y. USBee: Air-gap covert-channel via electromagnetic emission from USB. In Proceedings of the 2016 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 12–14 December 2016; pp. 264–268. [CrossRef]
24. Uzun, C.; Kahler, N.; de Peralta, L.G.; Kumar, G.; Bernussi, A.A. Programmable infrared steganography using photoinduced heating of nanostructured metallic glasses. In Proceedings of the 2017 Conference on Lasers and Electro-Optics (CLEO), San Jose, CA, USA, 14–19 May 2017; pp. 1–2.
25. Bieszczad, G.T.; Gogler, S.; Krupiński, M.; Ligienza, A.; Sawicki, K. The concept of thermovision sensor supporting the navigation of unmanned aerial platforms. *Meas. Autom. Monit.* **2019**, *65*, 15–18.
26. Bieszczad, G.; Kastek, M. Measurement of thermal behavior of detector array surface with the use of microscopic thermal camera. *Metrol. Meas. Syst.* **2011**, *18*, 679–690. [CrossRef]
27. Kohin, M.; Butler, N.R. Performance limits of uncooled VOx microbolometer focal plane arrays. In *Infrared Technology and Applications XXX*; Andresen, B.F., Fulop, G.F., Eds.; International Society for Optics and Photonics, SPIE: Orlando, FL, USA, 2004; Volume 5406, pp. 447–453. [CrossRef]
28. Olbrycht, R.; Więcek, B. New approach to thermal drift correction in microbolometer thermal cameras. *Quant. InfraRed Thermogr. J.* **2015**, *12*, 184–195. [CrossRef]
29. Nazdrowicz, J. Modelling Microbolometer Using Matlab/SIMULINK Package with Thermal Noise Sources. In Proceedings of the 2016 MIXDES—23rd International Conference Mixed Design of Integrated Circuits and Systems, Łódź, Poland, 23–25 June 2016; pp. 266–270. [CrossRef]