ISSN 1330-3651 (Print), ISSN 1848-6339 (Online) https://doi.org/10.17559/TV-20220417151954
Preliminary communication

Real-Time Building Management System Visual Anomaly Detection Using Heat Points Motion Analysis Machine Learning Algorithm

İsa AVCI*, Michael BIDOLLAHKHANI

Abstract: The multiplicity of design, construction, and use of IoT devices in homes has made it crucial to provide secure and manageable building management systems and platforms. Increasing security requires increasing the complexity of the user interface and the access verification steps in the system. Today, multi-step verification methods are used via SMS, call, or e-mail to do this. Another topic mentioned here is physical home security and energy management. Artificial intelligence and machine learning-based tools and algorithms are used to analyze images and data from sensors and security cameras. However, these tools are not always available due to the increase in data volume over time and the need for large processing resources. In this study, a new method is proposed to reduce the usage of process resources and the percentage of system error in anomaly detection by reducing visual data to critical points by using thermal cameras. This method can also be used in energy management using home and ambient temperature and user activity measurements. The statistical results of the visual comparison between the proposed method and the legacy CCTV-based visual and sensory surveillance shown in the results section demonstrate its reliability and accuracy.

Keywords: anomaly detection algorithm; building management system; digital image processing; internet of things; machine learning

1 INTRODUCTION

Today the Internet of Things (IoT) connected devices are the base of a paradigm that will be the network of smart devices and machines. Through IoT, these connected devices will be able to communicate with each other or the human clients (for monitoring or controlling goals). At the same time, during the growth of these revolutionary technologies being more and more in human daily life, serious issues will arise as to how to secure these systems from the network, software, and environmental attacks and errors [1]. There are several types of devices connected through IoT. Controlling these huge amounts of systems is not possible using human resources, therefore there are some technologies designed to act like sub-systems to manage and give access to the devices from users and the internet devices. A Building Management System (BMS) is one of the most important systems which is used in the management of devices connected over the same local network in a building. A BMS monitors, supervises, controls, and reports on smart building technology systems. These systems may include access control, video surveillance, fire alarms, Heating, Ventilation, and Air Conditioning (HVAC) control, programmable lighting, and electric power management. Such a considerable and huge number of IoT-based services most include anomaly detection systems (section) to decrease the risk of software or hardware misfunctioning, also detecting unusual activities to manage and secure the power, heating, ventilation, air-conditioning, physical access control, pumping stations, elevators, and lights [2]. Controlling the complexity to manage this huge amount of data, and securing the facilities is so vital. Also, many anomaly detection algorithms are not always available due to the wide variety of input data types and limited processing resources [3]. The complexity of anomaly detection tools should not make any delay and their accuracy should be high enough because any non-deterministic latency or false anomaly alert is not tolerable for the control system or security personnel [4]. To cover the critical and fundamental information about the building, the novel visual anomaly detection algorithm uses thermographic cameras for home surveillance which uses machine vision and intelligence techniques to process image frames taken from thermographic cameras.

Anomaly detection is referred to the identification of an odd, peculiar, or strange condition, situation, quality, etc., [5]. An anomaly is frequently misused with the meaning of only security threat, but the meaning of anomaly is considering so many other factors and the security threats. Anomaly detection in a BMS is to identify odd changes or strange conditions in sensory or image data entries. Examples of anomalies for a building may include human-based actions failure, robbery or breaking into the building without permissions, temperature/lighting control systems fault, and technology malfunctioning. Anomaly detection is such a vast distributed management system using visual and sensory data that may be a challenge for resource management and device communication aspects. Here the method is focusing on how to estimate the percentage of true anomaly detection using heat points motion analysis.

This study will follow the most recent and common methods for anomaly detection in building management systems using visual content processing. After a brief information about the latest advancement in this field, the proposed method implementation will be discussed. The results of the process will be demonstrated using a comparison between two other algorithms.

2 MATERIALS AND METHODS

Anomaly detection in camera images is usually comparing two or a sequence of images captured from a unique scene or environment and comparing them with each other. Difference estimation between visual data needs a complex and dynamic model definition for the new data to be compared with the last received one(s) [6, 7]. The modern machine intelligence or fuzzy systems-based comparison methods are concerned with fast, adaptive, and organizational learning which is along with memorizing information in vast amounts and active interaction and struggling with models and information retrieved from input devices [8]. These data transactions and process need more and more resources due to the increasing amount of data and sometimes after increasing more values because of adding new sensory or visual input devices. These kinds of systems are called Evolving Connectionist Systems (ECOSs). In this type of system, the framework assumes its structure and functionality should be evolved based on a new input stream of data [9, 10]. The difference between the reference image and retrieved images is determined as interference. Using these compare and time series anomaly detection techniques, if the pattern of the receiving new data is significantly distinct from reference data, the algorithm may detect an anomaly. Three approaches are sorting the comparing methods [11]. These are Full-Reference (FR), No-Reference (NR), and Reduce-Reference (RR).

- Full-Reference (FR),
- No-Reference (NR),
- Reduced-Reference (RR).

Almost comparing algorithms are following the FR approach, because they adopt statically modelled images as a reference to estimate variations from noises and cluttered scenes [12].

2.1 Full-Reference (FR)

The fully referenced approach for anomaly detection is to use the first input value as a reference to check if the next input value is changed based on that. Several visual anomaly detection methods use this approach, for instance [13]:

- One-class classification (OCSVM, SVDD, and OCCNN).
- Image reconstruction.
- Self-supervised classification.

To find out the abnormal changes in the sequence of entry data is common to use Machine learning algorithms. Time series-based data modelling is usually applied using two types of model generation techniques, Univariate and Multivariate time series modelling techniques. Autoregression (Including Autoregressive Integrated Moving Average (ARIMA) and Seasonal Autoregressive Integrated Moving Average (SARIMA)) and Facebooks Prophet methods are some instance algorithms.

Prophet algorithm, which was released by the Facebooks Core Data Science team, is an open-source library designed for automatic forecasting of univariate time series data. This is done based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects [14]. It works best with time series that have strong seasonal effects and several seasons of historical data. Prophet is robust to missing data and shifts in the trend, and typically handles outliers well [15].

Interference analysis of thermographic images was used before for estimation of error or multivariant distributions of input data in medicine [16, 17], geological science [18, 19], and some other fields, but it has never been used in building management systems anomaly estimation.

2.2 Proposed Method

2.2.1 Model Steps

The currently proposed method contains three thermal image processing steps, a model extraction method using Facebook time-series Prophet prediction, an information retrieval step, and two other steps for finalizing the result. The input data contains sensory and thermal video footage.

These data will be going to be processed as live stream data after some optimization steps and time-series-based data specifications classifying. The structural similarity will be estimated for each dual-frame to maintain the differences of each frame.

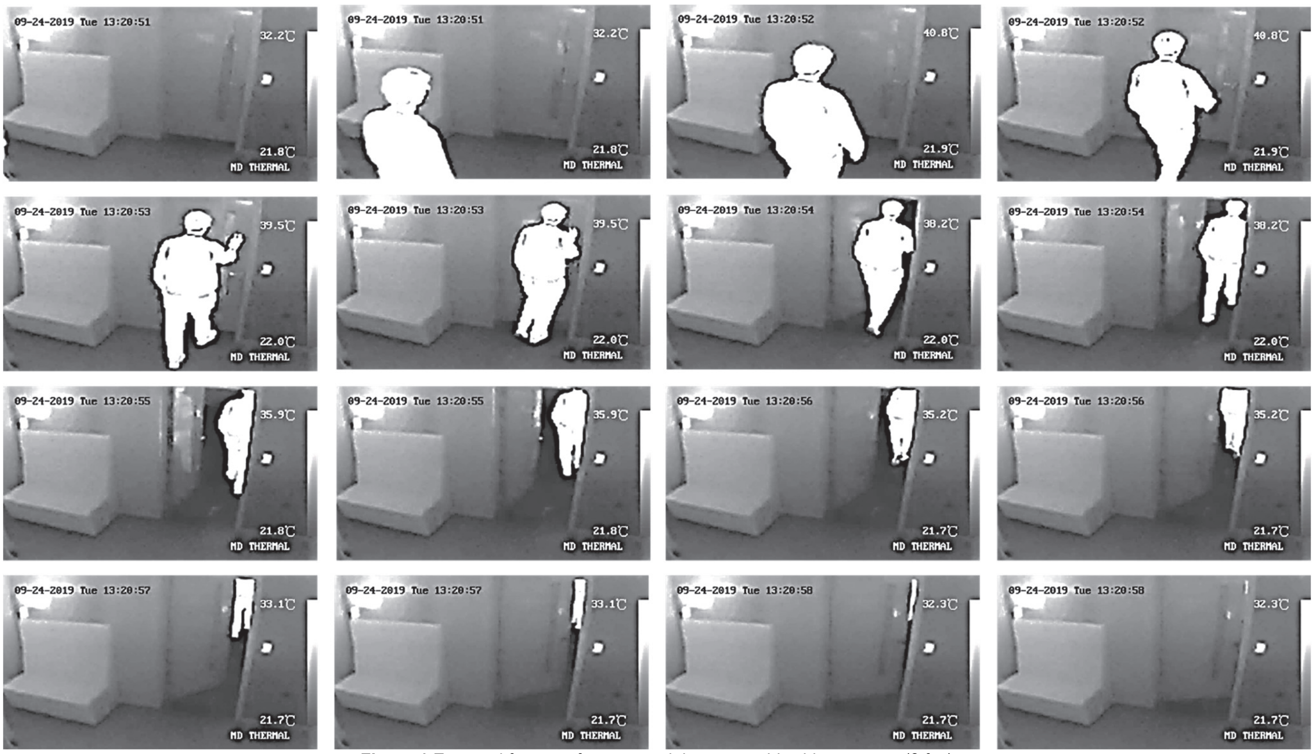

Figure 1 Exported frames of a 7-second thermographic video camera (2 fps)

Tab. 1 is a dataset of the Structural Similarity Index (SSIM) frame of data to show how the algorithm is handling mass data and computing the mean variance of the related time-series.

Table 1 SSIM dataset for day 1 to day n to estimate the most repetitive patterns and make the model

Day #1		…	Day #n		Mean Variance	
Time	SSIM		Time	SSIM	True value	Percentage
2019-09-24 13:20:49	0.0565	…	2019-10-24 13:20:49	0.0553	0.01186605	58.93749056
2019-09-24 13:20:49	0.0423		2019-10-24 13:20:49	0.0423	0.01016592	50.49311389
2019-09-24 13:20:50	0.0875		2019-10-24 13:20:50	0.0849	0.02013328	100
2019-09-24 13:20:50	0.1027		2019-10-24 13:20:50	0.1006	0.002054	24.35678059
2019-09-24 13:20:51	0.0284		2019-10-24 13:20:51	0.0273	0.00199367	23.64137427
2019-09-24 13:20:51	0.0193		2019-10-24 13:20:51	0.0189	0.00116088	13.76596857
2019-09-24 13:20:52	0.0193		2019-10-24 13:20:52	0.0190	0.00290955	34.50207934
2019-09-24 13:20:52	0.0107		2019-10-24 13:20:52	0.0102	0.00237512	28.16469168
2019-09-24 13:20:53	0.0104		2019-10-24 13:20:53	0.0102	0.00083488	9.900189376
2019-09-24 13:20:53	0.0072		2019-10-24 13:20:53	0.0068	0.00194778	23.09720063
2019-09-24 13:20:54	0.0068		2019-10-24 13:20:54	0.0066	0.00190428	22.58136813
2019-09-24 13:20:54	0.0093		2019-10-24 13:20:54	0.0089	0.0005631	6.67736278
2019-09-24 13:20:55	0.0009		2019-10-24 13:20:55	0.0008	0	0
2019-09-24 13:20:56	0.0033		2019-10-24 13:20:56	0.0033	0.00030411	3.606202797
2019-09-24 13:20:56	0.0159		2019-10-24 13:20:56	0.0153	0.00111713	13.24717152
2019-09-24 13:20:57	0.0235		2019-10-24 13:20:57	0.0233	0.00211914	25.12922493
2019-09-24 13:20:57	0.0401		2019-10-24 13:20:57	0.0381	0.00843297	100
2019-09-24 13:20:58	0.0456		2019-10-24 13:20:58	0.0437	0.00775472	100
2019-09-24 13:20:58	0.0727		2019-10-24 13:20:58	0.0720	0.00363975	52.42451216
2019-09-24 13:20:59	0.0578		2019-10-24 13:20:59	0.0578	0.00694284	100

2.2.2 Model Codes and Statistics

A big challenge of implementing such an algorithm which is processing the graphical data directly is to determine the resources consumption, where tuples and data frames came in handy. Also eliminating the cold areas which are not changing during at least two frames, helped to decrease RAM and GPU consumption, and helped to boost the algorithm's data handling speed. The codes of the proposed model are shown as Code-1.

```
#Variable to store Structural Similarity Index (SSIM)
df = pd.DataFrame(columns=['ds', 'ssim_score'])
df_list =[]
for i in range(0,20):#range(0,len(frame_list)-2):
# load the two input images
imageA = numpy.array(frame_list[i])#cv2.imread(frame_list[i])
imageB = numpy.array(frame_list[i+1])#cv2.imread(frame_list[i+1])
# convert the images to grayscale
grayA = cv2.cvtColor(imageA, cv2.COLOR_BGR2GRAY)
grayB = cv2.cvtColor(imageB, cv2.COLOR_BGR2GRAY)
# compute the Structural Similarity Index (SSIM) between the two
# images, ensuring that the difference image is returned
#pip install scikit-image
from skimage.metrics import structural_similarity as compare_ssim
(score, diff) = compare_ssim(grayA, grayB, full=True)
diff = (diff * 255).astype("uint8")
#Saving data to dataFrame gibi list
from datetime import date
today = date.today()
#Changing the similarity score into difference score.
df_list.append([i, 1-score])
print("SSIM of Frames({},{}): {}".format(i,i+1,score))
df =pd.DataFrame(df_list)
#Saving ssim results to csv file for further actions
df.to_csv('frames_ssim.csv', index=False,header=False)
```

Code 1 The class for Frames pre-processing and SSIM score calculation

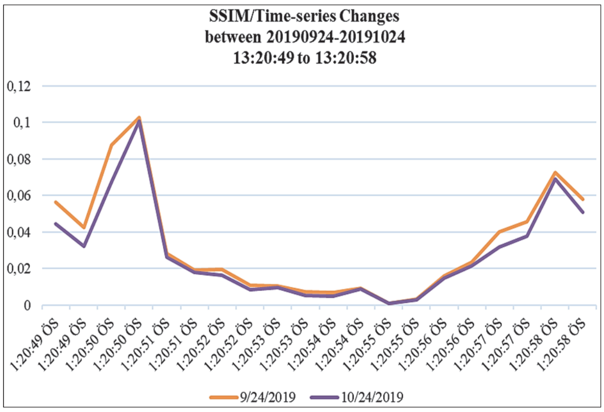

Figure 2 SSIM/Time-series changes

Variance estimation is an important step to find out the differences between the changes of every dual-frame during the closest time series. Extracted model based on the maximum variance percentage and time series is demonstrated using box-whisker plotting as in this Fig. 3. Having these data shows the usual high/low changes on the frames in Fig. 3, which will be considered for allowable detected changes.

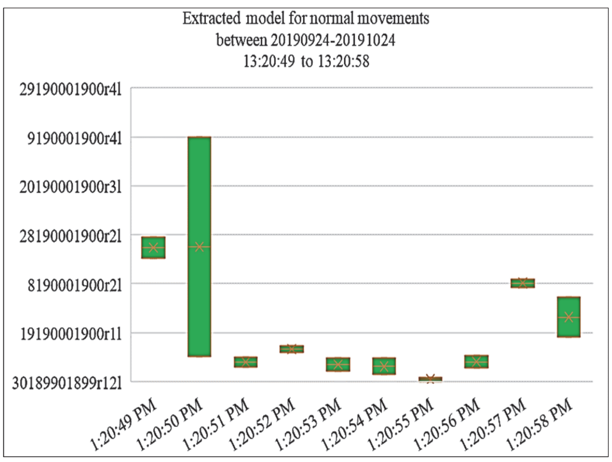

Figure 3 The extracted model for normal movements

By continuing this step and repeating the other time series the allowable variance of changes in frames will be found during the days of the week. The final segmentation to separate the models for each day of the week makes the models more flexible for holidays and specific events. For instance, during holidays and some special events which occasionally make some changes in variance they will be detectable by related models.

2.2.3 Flowchart of the Proposed Method

The survey method between different parts of the algorithm is drawn as a flowchart to show their relationships.

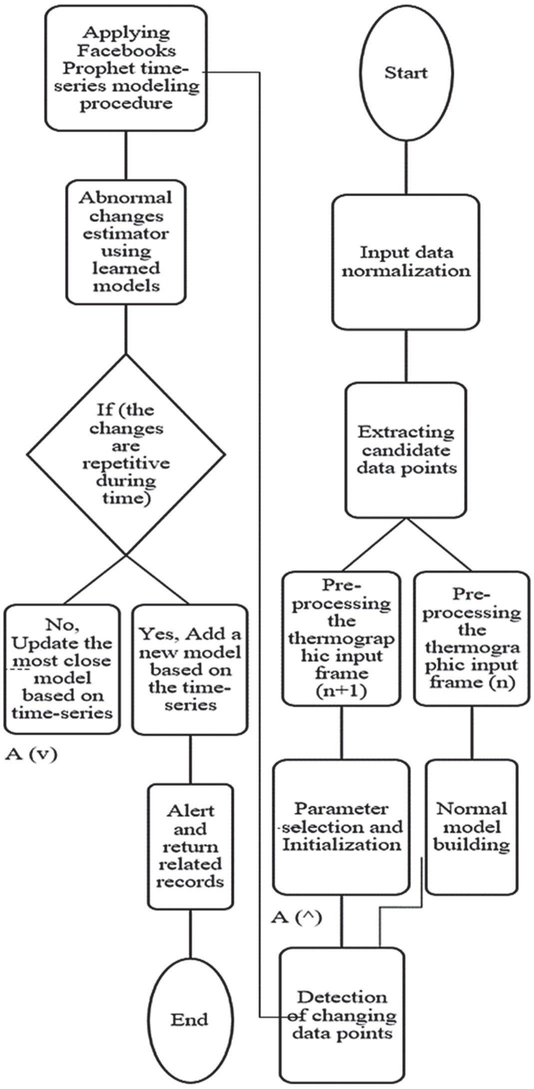

Figure 3 Flowchart of the proposed method

3 RESULTS

For instance, the final model extracted from data for Thursdays is demonstrated in Fig. 4 and Fig. 5. This model will be used to find out abnormal activities on the following Thursdays. Also, it is compatible to upgrade the weights and thresholds based on special events and changes in activity patterns.

The frame change threshold based on Structural Differences Indexing (SDIM) and time series are given for Thursday in Fig. 4. These data were obtained for a full day.

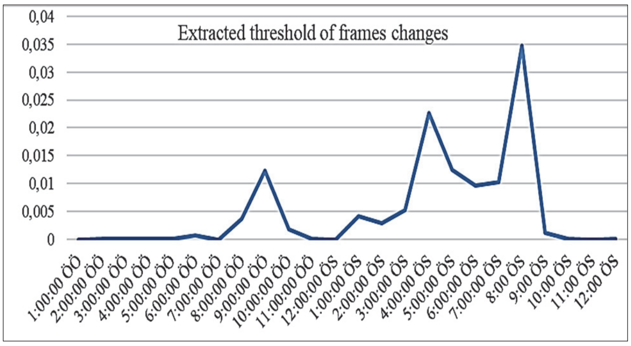

Figure 4 Extracted threshold of frames changes based on structural differences indexing (SDIM) and time-series

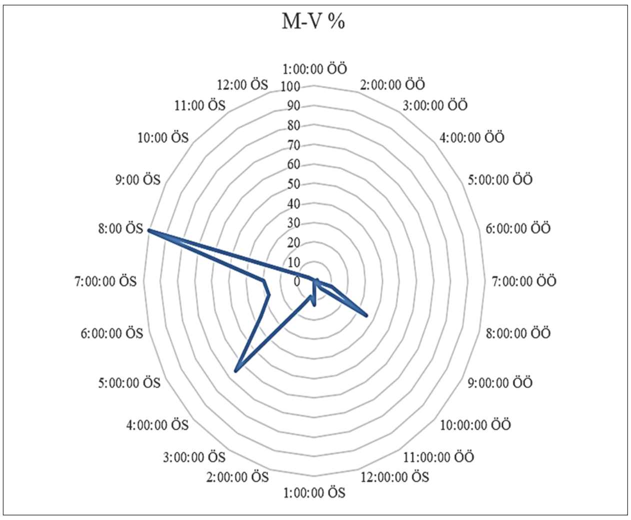

Figure 5 Estimated allowed activity based on time series per 24 hours of Thursdays

It usually shows the maximum activities on Thursdays, which are approximately at 09:00, 16:00, and 20:00 in Fig. 5. It means, that if the system sensed activities due to the schedule and allowed situations, there is no need to activate any special alert for an abnormal break or any error. Eq. (1) calculation of *TP*, *TN*, *FP*, *FN* refers to the number of True Positive, True Negative, False Positive, False Negative respectively also *Sensitivity*, *Specificity*, and *Accuracy* based on them [20].

$$Sensitivity = \frac{TP}{TP + FN} \tag{1}$$

$$Specificity = \frac{TN}{TN + FP} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{3}$$

where:
TP - number of true positive specimens; *FP* - number of false-positive specimens.

The accuracy of alerts on specific real suspicious situations is examined and compared to the other similar algorithms. In addition, it shows the comparison of over 75 data records along with the proposed model results in Tab. 2.

Table 2 Benchmarking over 75 data records

	Number of True Positives (*TP*)	Number of True Negatives (*TN*)	Number of False Positives (*FP*)	Number of False Negatives (*FN*)	Sensitivity	Specificity	False Alarm Rate (*FAR*)	Advantage
Ali Saglam et al. (A. Saglam and A. Temize, 2009)	40	0	35	0	53.33%	0%	100%	Lower Complexity
Yuan-Kai Wang et al. (Yuan-Kai Wang, Ching-Tang Fan, Ke-Yu Cheng, Peter Shaohua Deng, 2011)	56	16	2	1	98.24%	88.88%	11.11%	Lower False Alarm Rate (FAR)
Proposed algorithm	23	34	8	10	69.69%	80.95%	19.05%	Optimal FAR and Best redundancy removal

The testing videos captured from the analogue camera are digitized by using VGA resolution, 30 frames per second, and encoding with MPEG1. All the testing videos have contained 1800 to 2700 frames and at least 30 seconds of normal surveillance scenarios before the anomaly events happened, which is taking some algorithms that needed background modelling time into consideration. There are total 75 testing videos which are including 57 videos with anomaly events and 18 videos without any camera anomaly. An anomaly video has only one event inside.

4 CONCLUSIONS

Using received images from thermal cameras effectively gives high accuracy of detection of abnormal factors, such as the entry of people or other unusual events, such as a sudden rise or fall in temperature. Something else, as is mentioned, the very low error rate of the proposed algorithm compared to other existing deep learning-based methods. Another noteworthy point in this algorithm is its speed, which was not mentioned in the benchmark. The proposed method is based on data mining, which significantly increases its speed in obtaining flexible models from the received data. When comparing over 75 data records, it is seen that the proposed algorithm has the advantages of 19.05% Optimal FAR and Best redundancy removal. Facebook Prophet method helps the algorithm to find the key points to identify the patterns and prepare the final models, which itself has considerable speed, accuracy, and flexibility compared to time-series-based forecasting methods. It is common for these systems to be complex because of using deep neural networks for processing the big data entries, eliminating noise on data, and being flexible on human-defined values (not accurate and real-time), which presents new challenges and opportunities for the design and development of anomaly detection methods using visual data. Several characteristics of AI-based anomaly detection systems are discussed, as well as challenges in specifying acceptance criteria for such systems, such as accuracy, sensitivity, specificity, and false alarm rating. There are several challenges associated with testing this kind of system, including the environmental-based changes (day/night-time duration, season visual changes, moving obstacles on point of view, etc.), which make it difficult for testers to determine whether such a method has passed or failed by only a set of data from an environment. In addition to covering the testing of these systems across the life cycle, the researchers tested the algorithm in the outdoor and challenging environments and no significant changes in results were observed because of the advantages of the thermographic imaging methods.

5 REFERENCES

[1] Tsiknas, K., Taketzis, D., Demertzis, K., & Skianis, C. (2021). Cyber Threats to Industrial IoT: A Survey on Attacks and Countermeasures. *IoT*, *2*(1), 163-186. https://doi.org/10.3390/iot2010009

[2] Hentea, M. (2021). *Building an Effective Security Program for Distributed Energy Resources and Systems*. Wiley Professional, Reference & Trade.

[3] Sanz, C. (2003). *Computer delivered implicit vs. explicit feedback in processing instruction*. Mahwah, NJ: Erlbaum.

[4] Krutz, R. L. (2008). *Securing SCADA System.* Wiley India.

[5] Zimek, A. & Schubert, E. (2017). *Outlier Detection*. Springer, New York, NY. https://doi.org/10.1007/978-1-4899-7993-3_80719-1

[6] Amari, S. & Kasabov, N. K. (1998). *Brain-like Computing and Intelligent Information Systems*. Springer.

[7] Kasabov, N. (1998). Evolving fuzzy neural networks-algorithms, applications, and biological motivation. World Scientific, Singapore, 271-274.

[8] Kasabov, N. K. & Song, Q. (2002). DENFIS: dynamic evolving neural-fuzzy inference system and its application for time-series prediction. *IEEE Transactions on Fuzzy Systems*, *10*(2), 144-154. https://doi.org/ 10.1109/91.995117

[9] Bottu, L. & Vapnik, V. (1992). Local Learning Algorithms. *Neural Computation*, *4*(6), 888-900.

[10] Moody, J. & Darken, C. J. (1989). Fast Learning in Networks of Locally-Tuned Processing Units. *Neural Computation*, *1*(2), 281-294. https://doi: 10.1162/neco.1989.1.2.281

[11] Wang, Y. K., Fan, C. T., Cheng, K. Y., & Deng, P. S. (2011). Real-time camera anomaly detection for real-world video surveillance. *2011 International Conference on Machine Learning and Cybernetics*. https://doi: 10.1109/ICMLC.2011.6017032

[12] Gil-Jiménez, P., López-Sastre, R., Siegmann, P., Acevedo-Rodríguez, J., & Maldonado-Bascón, S. (2007). Automatic Control of Video Surveillance Camera Sabotage. *Lecture Notes in Computer Science*, *4528*. https://doi.org/10.1007/978-3-540-73055-2_24

[13] Yang, J., Xu, R., Qi, Z., & Shi, Y. (2021). Visual Anomaly Detection for Images: A Survey. https://doi.org/10.48550/arXiv.2109.13157

[14] Zunic, E., Korjenic, K., Hodzic, K., & Donko, D. (2020). Application of facebook's prophet algorithm for successful sales forecasting based on real-world data. https://doi.org/10.48550/arXiv.2005.07575

[15] Borneklint, N. (2021). Forecasting prices of Bitcoin and Google stock with ARIMA vs Facebook Prophet (Dissertation).

[16] Masko, M., Borowska, M., Domino, M. et al. (2021). A novel approach to thermographic images analysis of equine thoracolumbar region: the effect of effort and rider's body weight on structural image complexity. *BMC Vet Res 17*, *99*. https://doi.org/10.1186/s12917-021-02803-2

[17] Resendiz-Ochoa, E., Cruz-Albarran, I. A., Garduño-Ramon, M. A., Rodriguez-Medina, D. A., Osornio-Rios, R. A., & Morales-Hernández, L. A. (2021). Novel expert system to study human stress based on thermographic images. *Expert Systems with Applications*, *178*. https://doi.org/10.1016/j.eswa.2021.115024

[18] Cluni, F., Costarelli, D., Minotti, A. M., & Vinti, G. (2015). Enhancement of thermographic images as tool for structural analysis in earthquake engineering. *NDT & E International*, *70*, 60-72. https://doi.org/10.1016/j.ndteint.2014.10.001

[19] Costarelli, D., Cluni, F., Minotti, A. M., & Vinti, G. (2014). Applications of sampling Kantorovich operators to thermographic images for seismic engineering. https://doi.org/10.48550/arXiv.1411.2584

[20] Mousavi, S. S. Z., Zanjireh, M. M., & Oghbaie, M. (2020). Applying computational classification methods to diagnose Congenital Hypothyroidism: A comparative study. *Informatics in Medicine Unlocked*, *18*, 100281. https://doi.org/10.1016/j.imu.2019.100281

[21] Saglam, A. & Temizel, A. (2009). Real-time adaptive camera tamper detection for video surveillance. *2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance*. https://doi: 10.1109/AVSS.2009.29

Contact information:

İsa AVCI, Assistant Professor, PhD
(Corresponding author)
Karabuk University,
Kastamonu Yolu Demir Çelik Kampüsü, 78050 Kılavuzlar/Karabük
E-mail: isaavci@karabuk.edu.tr

Michael BIDOLLAHKHANI, Student, MsC
Karabuk University,
Kastamonu Yolu Demir Çelik Kampüsü, 78050 Kılavuzlar/Karabük
E-mail: michaelbidollahkhani@ogrenci.karabuk.edu.tr