JCSE Journal of Computing Science and Engineering

Regular Paper
Journal of Computing Science and Engineering,
Vol. 17, No. 4, December 2023, pp. 207-215

Handling Imbalanced Data Using a Cascade Model for Image-Based Human Action Recognition

Wahyono*, Suprapto, Adam Rezky, and Nur Rokhman
Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia
wahyo@ugm.ac.id, sprapto@ugm.ac.id, adamrezky2020@mail.ugm.ac.id, nurrokhman@ugm.ac.id

Kang-Hyun Jo
Department of Electrical Engineering, University of Ulsan, Ulsan, Korea
acejo@ulsan.ac.kr

Abstract
Human action recognition plays a crucial role in intelligent monitoring systems, which are based on analyzing the possibility of anomalous events related to human behavior, such as theft, fights, and other incidents. However, by definition, anomalous events occur somewhat infrequently, thus leading to small and unbalanced data compared to data on other events. Such a data imbalance causes the human action recognition model to fail to produce optimal accuracy. To overcome the problem of imbalanced data, the typical methods used are oversampling and undersampling. However, these two methods are not considered to be very effective, because they cause the loss of a significant amount of information or deviations from reality. Therefore, the current paper proposes a cascade modeling strategy to address data imbalance problems, particularly in the case of human action recognition. The proposed strategy consists of several steps: preprocessing, feature extraction, modeling, and evaluation. The BAR dataset experiment found that the cascade model outperformed the other examined methods with an accuracy of 56.38%. However, there is still potential for further improvement through continued research.

Category: Computer Graphics / Image Processing

Keywords: Human action recognition; Imbalanced data; Cascade modeling; HOG feature extraction; Support vector machine

I. INTRODUCTION

Human action recognition is pivotal in developing computer vision systems, which aim to identify and classify human movements from videos or series of images [1, 2]. A significant challenge in action recognition is data imbalance, where the number of samples between different classes is uneven in the dataset. This phenomenon can impact the accuracy and performance of the model, leading to bias toward the majority class and neglect of the minority class [3, 4]. Data imbalance in action recognition can lead to a model to recognize fewer common actions represented in the dataset than it would otherwise. In this scenario, models tend to prioritize the majority class as they disregard crucial but less frequent action variants. This bears serious implications in practical

Open Access http://dx.doi.org/10.5626/JCSE.2023.17.4.207 http://jcse.kiise.org

Received 20 November 2023; **Accepted** 17 December 2023
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

applications, where diversity of actions is crucial for understanding different contexts and situations. A common problem arising from imbalanced data is a disparity in the representation of action classes, which makes it difficult for the model to generate accurate predictions for minority classes and diminishes the overall reliability of the action recognition system. These limitations threaten such a model's ability to address important real-world situations that involve infrequently occurring actions. Such a model also overlooks the context and subtleties of minority actions, thus resulting in substantial inaccuracies in those classes [5].

There are several general methods used to overcome the problem of imbalanced data. First, a frequently used approach is oversampling, where the number of samples from the minority class is increased by duplicating or creating synthetic samples [6-8]. A popular oversampling technique is the synthetic minority over-sampling technique (SMOTE), which involves generating synthetic samples based on relationships between existing minority samples [7]. The second method is undersampling, where the number of samples from the majority class is reduced to balance it with the minority class. Random undersampling and Tomek links are commonly used undersampling methods [9]. Oversampling and undersampling can also be combined to obtain the advantages of both strategies [10]. However, one significant weakness of the oversampling techniques lies in the potential risk of overfitting. When synthetic samples are generated to augment the minority class, the model may become too specialized in capturing the nuances of the training data, thus leading to poor generalization on unseen instances. Oversampling can also substantially increase computational complexity and memory requirements, particularly when the dataset is already extensive. Creating synthetic samples may introduce noise and unrealistic patterns, thereby adversely affecting the model's ability to discern genuine features and hindering its performance on real-world data. On the other hand, while addressing a class imbalance, undersampling comes with its own weaknesses. One primary concern is the loss of information. In particular, reducing the number of majority class samples may lead to a dataset that inadequately represents the true characteristics and diversity of the majority class, thus limiting the model's ability to learn essential patterns. Undersampling may also increase the model's sensitivity to outliers within the remaining majority class samples, which could potentially result in a less robust and more biased model. The challenge is to balance addressing the class imbalance while retaining sufficient information for the model to generalize effectively.

Further, data imbalances can often be addressed more effectively by placing the focus on how the model is built rather than changing the existing data. Changing the data, such as through oversampling or undersampling, can create the risk of overfitting or losing valuable information. It is better to instead adjust the model to understand and address such imbalances. Strategies such as assigning class weights, using appropriate evaluation metrics such as F1-score, or applying special algorithm techniques, such as cascade modeling, can also help models better handle imbalanced class distributions [11].

Therefore, this research proposes the use of a cascade modeling approach to overcome data imbalance. Cascade modeling is an ensemble method wherein a series of machine learning models is built sequentially. Each model in the early stages focuses on detecting and classifying minority classes, while the models in the later stages concentrate on overcoming the remaining difficulties. This approach allows the model to gradually improve its skill at handling imbalances, thus reducing the risk of overfitting.

The rest of this paper is organized as follows: Section II describes the proposed method; Section III presents the experimental setting, results, and discussion; and Section IV concludes our work.

II. THE PROPOSED METHOD

A. Method Overview

This research method comprises several key steps, as illustrated in Fig. 1. First, to ensure image quality and consistency, the data undergoes preprocessing that includes resizing, enhancing, and smoothing. Next, feature extraction is performed using a histogram of oriented gradients (HOG) [12] to extract crucial information regarding the orientation and distribution of light-intensity gradients in human action images. Next, modeling is applied using the cascade model to recognize and represent action patterns hierarchically. The final step involves validation with k-fold cross-validation to assess the overall performance of the model and measure its reliability in recognizing human actions. Through this approach, this methodology can offer an accurate and robust representation of variations in the dataset, thereby enabling a reliable analysis of various human actions. The details involved in each step are provided in the next subsection.

B. Dataset Characteristics

In the experiment, a public dataset was used to evaluate the effectiveness of the proposed method; in particular, we employed the Biased Action Recognition (BAR) dataset [13]. The BAR dataset comprises authentic images depicting six distinct action classes associated with specific locations. The selection of these six action classes was meticulously conducted by referencing imSitu [14]. To elaborate, our criteria involved choosing action classes where the images for each candidate action shared common

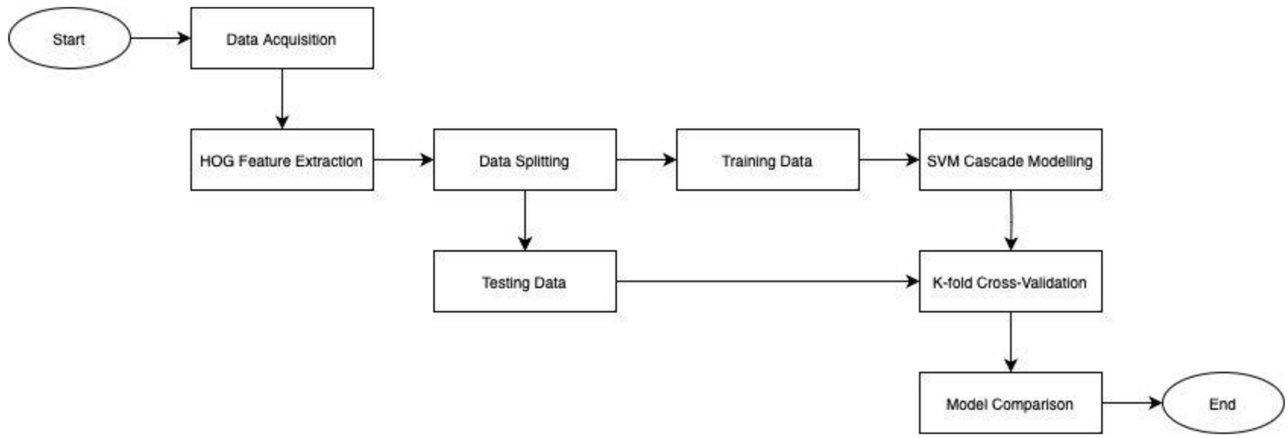

Fig. 1. Pipelines of the proposed method pipelines.

Fig. 2. Example images from datasets in this research [13] (https://github.com/alinlab/BAR).

Table 1. Distribution of dataset for training and testing

Class	Train set	Test set	Total
Climbing	260	66	326
Diving	416	104	520
Fishing	130	33	163
Racing	268	68	336
Throwing	253	64	317
Vaulting	223	56	279
Total	1,550	391	1,941

environmental characteristics. It was also important to ensure that the environmental features of the action class candidates were sufficiently distinctive to facilitate action classification based solely on the place attributes. Consequently, we identified six prototypical action-place pairs as follows and as shown in Fig. 2: (Climbing, RockWall), (Diving, Underwater), (Fishing, WaterSurface), (Racing, APavedTrack), (Throwing, PlayingField), and (Vaulting, Sky). The dataset was then divided into training and testing with a respective ratio of 80%:20%, as indicated in Table 1.

The main reason that the BAR dataset was chosen is that it is publicly available and has been involved in many studies, which will simplify the evaluation and comparison process. The BAR dataset also provides six different classes where candidate actions share commonplace characteristics. These place characteristics of action class candidates are distinct and allow for actions to be classified solely from place attributes.

C. Preprocessing

Preprocessing is the initial step in this research, and it is used to enhance image quality with the aim of improving the model's ability to capture essential features. This process involves two main stages: resizing and smoothing using the Gaussian smoothing method. The image size is first adjusted through resizing to ensure consistency and uniformity in the analysis. Meanwhile, Gaussian smoothing is applied to reduce noise and refine the image structure, thus ensuring a clearer and more reliable visual representation [15]. It is anticipated that the combination of these two preprocessing techniques will lead to a more optimal resulting image for further feature extraction, thus strengthening the basis for an in-depth analysis of the image data used in this research.

D. Feature Extraction

This research adopts a feature extraction method in the form of HOG [12] to analyze and represent the form of human action in each context. HOG is a feature extraction technique that effectively identifies oriented patterns and structures, and it has proven valuable in understanding human body movement. The primary advantage of HOG

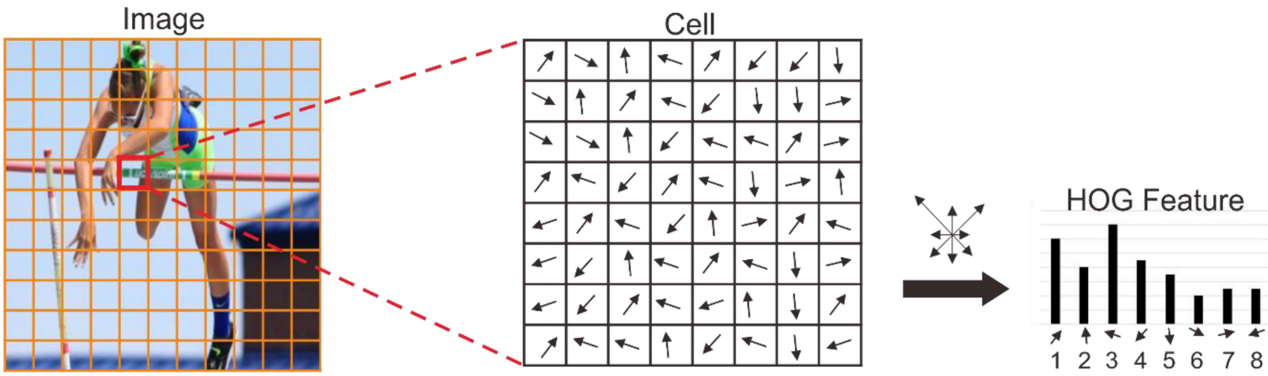

Fig. 3. Illustration of HOG feature extraction for action recognition.

lies in its ability to capture critical details in action, particularly those related to the orientation and distribution of light-intensity gradients in the image. This method utilizes this information to construct histograms reflecting significant changes in texture and contour in images, ultimately resulting in a rich representation of essential features related to human action. In the context of human movement analysis, HOG can offer significant advantages by capturing changes in shape and orientation that occur during an action. For example, when a person performs a characteristic hand or foot movement, HOG can represent that change in shape through the distribution of light-intensity gradients, thus allowing the system to recognize and understand the action. HOG's accuracy and robustness in handling lighting variations, background changes, and shooting angle variations make it a reliable choice for the analysis of human movement occurring in various conditions.

In this study, the HOG configuration uses orientation with 9 bins, which determines that the histogram gradient will have nine orientation bins. This implies that the gradient direction in each cell will be divided into eight different directions. Second, pixel_per_cell is set at (16,16), so the size of each cell is set to 16×16 pixels. The histogram gradient will be calculated in a 16×16-pixel grid across the image. Third, cells_per_block (1,1) indicates that each block (for histogram normalization) will consist of 1×1 cells. This means there is no additional grouping of cells within a block, and that each cell will be considered its block. Fourth, visualize True indicates that the function will also return the HOG visualization image and the HOG feature vector. This is useful for understanding how HOG represents images. Finally, multichannel=False indicates that the input image is grayscale and not a multichannel image (like RGB). Overall, this configuration determines how HOG features are calculated and represented from a given image, with a focus on gradient orientation, cell size, block normalization, visualization, and image channel type. Fig. 3 illustrates how the HOG features are extracted.

E. Constructing Cascade Model

There are at least 720 possible cascade models in a dataset with six classes. However, it would be inefficient to attempt to train all possible cascade models individually. Therefore, in this research, we propose the formation of a cascade model based on the number of datasets used in training. To form this cascade model, we will train using binary classification, because the proposed cascade model is binary. The formation of the cascade model begins by dividing the training data containing six classes into two class groups, thus ensuring the presence of a balanced amount of data in each group. The subsequent iteration involves dividing each class group into two class subgroups according to the same rules. This process continues until each formed class group only contains one target class. Next, we train each pair of class groups from this cascade model, resulting in five binary classification models in the experiment, as shown in Fig. 4. Each model is then trained using a support vector machine (SVM).

SVMs offer several benefits that have seen them come to be widely employed in machine learning tasks [16]. One key advantage is their effectiveness in handling high-dimensional data and complex classification problems. SVMs excel in scenarios where the data is not linearly separable by transforming it into a higher-dimensional space using a kernel trick, which allows for the identification of nonlinear patterns. SVMs are also robust in the presence of outliers, as they focus on defining a decision boundary by maximizing the margin between classes, thereby enhancing their generalization to new, unseen data. The kernel's flexibility, ability to handle linear and nonlinear relationships, and resilience to outliers contribute to the versatility of SVMs as well as their success in various applications, such as image recognition, text classification, and bioinformatics. The evaluation metric used is accuracy, which calculates how well the model classifies data. Therefore, in our research, we use SVM in cascade modeling.

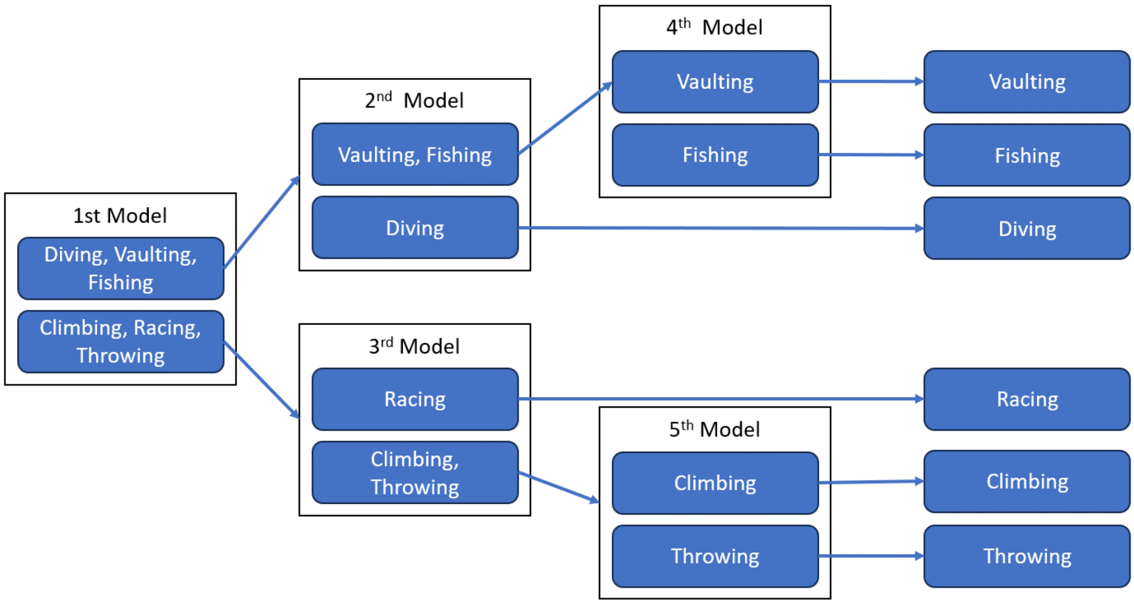

Fig. 4. Proposed cascade model based on the amount of data.

III. RESULTS AND DISCUSSION

A. Experiment Setting

This section describes the experimental settings and evaluates the influences of several parameters in the cascade model. The model was implemented using the Python programming language with the OpenCV library running on MacOS Ventura 13.0.1. The system has 8 GB LPDDR4 RAM, an 8-core Apple M1 chip running at 2.10 GHz, and an 8-core Apple M1 GPU.

B. The Effect of Parameter Tuning in Feature Extraction

In our analysis of the HOG feature extraction, we investigated the impact of varying cell sizes and orientation bins on the accuracy of human action recognition. Table 2 presents the comparison of two different cell sizes, 8×8 and 16×16 pixels, each with two levels of orientation bins, 9 and 18, respectively. The results show that there is a marginal difference in accuracy across these configurations.

Table 2. Effect of HOG parameter tuning

Cell size (pixels)	Orientation bins	Accuracy (%)
8×8	9	55.95
	18	55.92
16×16	9	56.12
	18	56.08

In particular, the accuracy of the 8×8 cell size was 55.95% with 9 orientation bins, and it slightly decreased to 55.92% with 18 bins. Similarly, for the 16×16 cell size, the accuracy was marginally higher at 56.12% for 9 bins, and it slightly decreased to 56.08% for 18 bins. These findings suggest that while the cell size and the number of orientation bins impact accuracy, the variations are not significant in the context of our dataset.

This outcome implies that finer granularity in feature extraction only sometimes translates into significant improvements in accuracy. A possible determining factor here is that, while smaller cells and a greater number of bins capture more detailed information, they may also introduce noise or irrelevant features into the classification process, particularly in a dataset where actions may not exhibit drastically different gradient orientations across varied actions.

The comparable performance of the larger cell size with fewer orientation bins suggests the existence of some threshold, beyond which additional detail does not yield better classification. This points toward an optimal balance where sufficient feature detail is captured without overcomplicating the model, thereby avoiding overfitting while maintaining computational efficiency. These results emphasize the importance of context-specific tuning of HOG parameters. In datasets where action patterns are not distinctly varied in terms of orientation and gradient distribution, increasing the granularity in HOG parameters is likely to only have marginal benefits. Thus, our results emphasize the need for a tailored approach in setting HOG parameters that align with the specific characteristics and complexities of the dataset at hand.

C. The Effect of Parameter Tuning in the SVM Model

Regarding the SVM model, we investigate the effects of different kernel types and the penalty parameter C on classification accuracy. As can be seen in Table 3, the kernels tested are Linear, Polynomial, and radial basis functions (RBF), each with values at 1, 5, and 10 C. The Linear kernel showed a slight increase in accuracy from 56.11% at C = 1 to 56.23% at C = 5, followed by a slight decrease to 56.10% at C = 10. The accuracy of the Polynomial kernel gradually increased from 56.15% at C = 1 to 56.38% at C = 5, and then it decreased slightly to 56.12% at C = 10. The RBF kernel presented consistent accuracy levels, starting from 56.09% at C = 1, slightly increasing to 56.18% at C = 5, and maintaining at 56.10% at C = 10.

These results reveal an informative and subtle pattern with certain implications. With the Linear kernel, a modest increment in accuracy was observed as C increased from 1 to 5, suggesting that a moderate penalty on misclassifications improves model performance. However, further increasing C to 10 resulted in a marginal decrease in accuracy, potentially indicating an onset of overfitting whereby the model becomes too strict in fitting the training data, thereby losing generalization capability.

In the case of the Polynomial kernel, a similar trend was noted, with a gradual rise in accuracy peaking at C = 5 and followed by a minor drop at C = 10. This trend could reflect the Polynomial kernel's sensitivity to the balance between model complexity and overfitting, particularly in datasets involving intricate patterns, such as those capturing human actions. On the other hand, the RBF kernel presented consistent and not markedly varying accuracy across different C values. This consistency might be attributable to the RBF kernel's inherent ability to effectively manage nonlinear data separation, thus making it less sensitive to changes in the penalty parameter within the tested range.

These results collectively suggest that, while the choice of kernel and the C value influence the SVM's accuracy, it is possible that these impacts are particularly pronounced in our action recognition context. This indicates that the chosen dataset, with its specific characteristics and complexities, may not be extraordinarily sensitive to these hyperparameters within the tested ranges. This underscores the importance of taking a balanced approach when selecting hyperparameters, as a high penalty could lead to overfitting whereas a low penalty might result in underfitting.

Table 3. Effect of SVM parameter tuning

Kernel	C value	Accuracy (%)
Linear	1	56.11
	5	56.23
	10	56.10
Polynomial	1	56.15
	5	**56.38**
	10	56.12
RBF	1	56.09
	5	56.18
	10	56.10

The bold font indicates the best performance in each test

D. State-of-the-Art Comparison

We examine how our cascading classifier SVM model with HOG feature extraction compares against recent advancements in the field of action recognition, as presented in [17] and Table 4. Our methodology achieved an accuracy of 56.38%. Ahmed et al. [18] implemented OccamResNet with PGI and achieved an accuracy of 55.90%, while Sagawa et al. [19] utilized OccamResNet with gDRO and achieved an accuracy of 52.9%. Pezeshki et al. [20] applied a ResNet model with stochastic depth and achieved an accuracy of 51.3%. This comparison emphasizes the efficacy of our approach, which aligns with the trend of integrating advanced feature extraction with using robust classification models.

The observed marginal improvement in the accuracy of the methodology, which was considered to be primarily driven by the integration of HOG feature extraction with a

Table 4. State-of-the-art comparison

Study	Year	Methodology	Accuracy (%)
Our proposed method	2023	Cascade SVM model with HOG feature extraction	56.38
Shrestha et al. [17]	2022	OccamResNet	52.60
Ahmed et al. [18]	2019	OccamResNet with PGI	55.90
Sagawa et al. [19]	2019	OccamResNet with gDRO	52.90
Pezeshki et al. [20]	2020	ResNet with SD	51.30

PGI: predictive group invariance, gDRO: group distributionally robust optimization, SD: stochastic depth.

Table 5. Training time comparison

Methodology	Accuracy (%)	Time (ms)
Cascade modeling	56.38	2,403.47
Non-cascade modeling	52.47	700.95

cascading SVM, warrants a deeper examination to understand its underlying efficacy. This slight improvement in performance can be ascribed to several key factors in the methodology's design and implementation.

First, the HOG feature extraction plays a pivotal role by focusing on the orientation and intensity gradients within the images. HOG effectively captures the edge and shape information, which is crucial in distinguishing different human actions. This method outlines the form and posture inherent in action recognition tasks, and it translates intricate visual details into a robust and descriptive feature set. The granularity and precision of HOG in capturing these details provide the classifier with highly informative input.

The cascading SVM model builds upon the detailed features, in contrast to standard classification approaches, and the cascading model adds a hierarchical decision-making process, where multiple SVM classifiers are arranged in a cascading structure. Each classifier in this cascade specializes in differentiating specific aspects or categories of actions, thus making the overall system more adept at handling human actions' complex and varied nature. This hierarchical approach leads to a more nuanced and refined classification process in which decisions are made progressively, ultimately leading to higher overall accuracy.

The synergy between the detailed feature representation provided by HOG and the nuanced decision-making capability of the cascading SVM classifiers results in a more effective recognition of human actions. This combination harnesses the strengths of both feature extraction and classification, thus enabling the system to discern subtle differences between actions that other models might miss.

Moreover, even though the resulting cascade model exhibits better accuracy than other methods, its training process takes nearly four times longer when compared to the non-cascade model, as indicated in Table 5. This is because training on cascade models typically requires more models to be trained compared to training without a cascade. This issue could certainly pose a challenge in the future; parallel processing can be implemented in future research to address this problem.

IV. CONCLUSION

The problem of imbalanced data in the case of human action recognition in intelligent monitoring systems is a result of the fact that many human actions occur infrequently in the monitoring area, which is particularly true for classes that are anomalous events. Thes present research overcame the imbalanced data problem by proposing a cascade modeling strategy based on SVMs and histograms of oriented gradients. The proposed strategy was evaluated using BAR and resulted in an accuracy of 56.38%. These results compare favorably with the methods presented by Ahmed et al. [18], Sagawa et al. [19], and Pezeshki et al. [20], which resulted in accuracies of 55.90%, 52.90%, and 51.30%, respectively. However, there are still several directions for future research to explore. First, this strategy still needs to achieve further improvements in accuracy, such as by using a hybrid strategy to overcome imbalanced datasets. Second, although cascade modeling produces better accuracy, the training process still takes quite a long time. Therefore, parallel processing in cascade modeling can be considered.

Conflict of Interest(COI)

The authors have declared that no competing interests exist.

ACKNOWLEDGMENTS

This research was funded by the Department of Computer Science and Electronics, Universitas Gadjah Mada, Yogyakarta, Indonesia, under Research Laboratory Capacity Grant Year 2023 (No. 120/UN1/FMIPA.1.3/TU/PT.01.03/2023).

REFERENCES

1. G. A. S. Surek, L. O. Seman, S. F. Stefenon, V. C. Mariani, and L. D. S. Coelho, "Video-based human activity recognition using deep learning approaches," *Sensors*, vol. 23, no. 14, article no. 6384, 2023. https://doi.org/10.3390/s23146384
2. A. Sanchez-Caballero, D. Fuentes-Jimenez, and C. Losada-Gutierrez, "Real-time human action recognition using raw depth video-based recurrent neural networks," *Multimedia Tools and Applications*, vol. 82, no. 11, pp. 16213-16235, 2023. https://doi.org/10.1007/s11042-022-14075-5
3. J. M. Johnson and T. M. Khoshgoftaar, "Survey on deep learning with class imbalance," *Journal of Big Data*, vol. 6, article no. 27, 2019. https://doi.org/10.1186/s40537-019-0192-5
4. P. Kumar, R. Bhatnagar, K. Gaur, and A. Bhatnagar, "Classification of imbalanced data: review of methods and applications," *IOP Conference Series: Materials Science and Engineering*, vol. 1099, article no. 012077, 2021. https://doi.org/10.1088/1757-899X/1099/1/012077
5. S. J. Basha, S. R. Madala, K. Vivek, E. S. Kumar, and T. Ammannamma, "A review on imbalanced data classification techniques," in *Proceedings of 2022 International Conference on Advanced Computing Technologies and*

Applications (ICACTA), Coimbatore, India, 2022, pp. 1-6. https://doi.org/10.1109/ICACTA54488.2022.9753392

6. F. Rodriguez-Torres, J. F. Martinez-Trinidad, and J. A. Carrasco-Ochoa, "An oversampling method for class imbalance problems on large datasets," *Applied Sciences*, vol. 12, no. 7, article no. 3424, 2022. https://doi.org/10.3390/app12073424
7. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of Artificial Intelligence Research*, vol. 16, pp. 321-357, 2002. https://doi.org/10.1613/jair.953
8. P. Gnip, L. Vokorokos, and P. Drotar, "Selective oversampling approach for strongly imbalanced data," *PeerJ Computer Science*, vol. 7, article no. e604, 2021. https://doi.org/10.7717/peerj-cs.604
9. T. Elhassan, M. Aljurf, F. Al-Mohanna, and M. Shoukri, "Classification of imbalance data using Tomek link (t-link) combined with random under-sampling (RUS) as a data reduction method," *Global Journal of Technology & Optimization*, vol. 7, no. S1, 2016. https://doi.org/10.4172/2229-8711.s1111
10. E. F. Swana, W. Doorsamy, and P. Bokoro, "Tomek link and SMOTE approaches for machine fault classification with an imbalanced dataset," *Sensors*, vol. 22, no. 9, article no. 3246, 2022. https://doi.org/10.3390/s22093246
11. B. Krawczyk, "Learning from imbalanced data: open challenges and future directions," *Progress in Artificial Intelligence*, vol. 5, pp. 221-232, 2016. https://doi.org/10.1007/s13748-016-0094-0
12. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proceedings of 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, San Diego, CA, USA, 2005, pp. 886-893. https://doi.org/10.1109/CVPR.2005.177
13. J. Nam, H. Cha, S. Ahn, J. Lee, and J. Shin, "Learning from failure: de-biasing classifier from biased classifier," *Advances in Neural Information Processing Systems*, vol. 33, pp. 20673-20684, 2020.
14. M. Yatskar, L. Zettlemoyer, and A. Farhadi, "Situation recognition: Visual semantic role labeling for image understanding," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Las Vegas, NV, USA, 2016, pp. 5534-5542. https://doi.org/10.1109/CVPR.2016.597
15. J. P. D'Haeyer, "Gaussian filtering of images: a regularization approach," *Signal Processing*, vol. 18, no. 2, pp. 169-181, 1989. https://doi.org/10.1016/0165-1684(89)90048-0
16. M. A. Hearst, S. T. Dumais, E. Osuna, J. Platt, and B. Scholkopf, "Support vector machines," *IEEE Intelligent Systems and their Applications*, vol. 13, no. 4, pp. 18-28, 1998. https://doi.org/10.1109/5254.708428
17. R. Shrestha, K. Kafle, and C. Kanan, "OccamNets: mitigating dataset bias by favoring simpler hypotheses," in *Computer Vision – ECCV 2022*. Cham, Switzerland: Springer, 2022, pp. 702-721. https://doi.org/10.1007/978-3-031-20044-1_40
18. F. Ahmed, Y. Bengio, H. Van Seijen, and A. Courville, "Systematic generalisation with group invariant predictions," in *Proceedings of the 9th International Conference on Learning Representations (ICLR),* Virtual Event, Austria, 2021.
19. S. Sagawa, P. W. Koh, T. B. Hashimoto, and P. Liang, "Distributionally robust neural networks for group shifts: on the importance of regularization for worst-case generalization," 2019 [Online]. Available: https://arxiv.org/abs/1911.08731.
20. M. Pezeshki, O. Kaba, Y. Bengio, A. C. Courville, D. Precup, and G. Lajoie, "Gradient starvation: a learning proclivity in neural networks," 2020 [Online]. Available: https://arxiv.org/abs/2011.09468.

Wahyono https://orcid.org/0000-0002-2639-8411

Wahyono received the B.Sc. degree in Computer Science from Universitas Gadjah Mada, Indonesia, and the Ph.D. degree from The University of Ulsan, South Korea. He is an Associate Professor at the Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia. His research interests include machine learning, computer vision, and pattern recognition. He actively participates as a member of the societies, a reviewer in reputable international journals, and an editor in several journals.

Suprapto

Suprapto received the B.Sc. degree in Computer Science from Universitas Gadjah Mada, Indonesia, the master's degree from the University of Indonesia, and the Ph.D. degree from the Universitas Gadjah Mada, Indonesia. He is an Associate Professor at the Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia. His research interests include algorithm analysis and design, simulation, computational logic, graph theory, and pattern recognition.

Adam Rezky

Adam Rezky received the B.Sc. degree in Computer Science from Universitas Gadjah Mada, Indonesia. He is active as a research and teaching assistant at the Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia. His research interests include machine learning, computer vision, pattern recognition, and intelligent systems.

Nur Rokhman https://scholar.google.co.id/citations?user=W58AawoAAAAJ

Nur Rokhman received the B.Sc., Master, and Ph.D. degrees in Computer Science from Universitas Gadjah Mada, Indonesia. He is an Associate Professor at the Department of Computer Science and Electronics, Universitas Gadjah Mada, Indonesia. His research interests include Computation, Information Systems, and outlier detection. He actively participates as a member of societies and a reviewer in reputable international journals.

Kang-Hyun Jo https://orcid.org/0000-0001-8317-6092

Kang-Hyun Jo received the Ph.D. degree in computer-controlled machinery from Osaka University, Osaka, Japan, in 1997. After a year of experience with the Electronics and Telecommunications Research Institute (ETRI), Daejeon, South Korea, as a Postdoctoral Research Fellow, he joined the School of Electrical Engineering, University of Ulsan, Ulsan, South Korea. He is currently the Faculty Dean of the School of Electrical Engineering, University of Ulsan, Ulsan, South Korea. His research interests include computer vision, robotics, autonomous vehicles, and ambient intelligence. He has served as the Director or an AdCom Member for the Institute of Control, Robotics and Systems, The Society of Instrument and Control Engineers, the IEEE IES Technical Committee on Human Factors Chair, an AdCom Member, and the Secretary until 2019. He has also organized many international conferences, such as the International Workshop on Frontiers of Computer Vision, the International Conference on Intelligent Computation, the International Conference on Industrial Technology, the International Conference on Human System Interactions, and the Annual Conference of the IEEE Industrial Electronics Society. He is also an Editorial Board Member for international journals, such as the International Journal of Control, Automation, and Systems and Transactions on Computational Collective Intelligence.