Building a People Counting Solution Using Computer Vision

Favour Olaoye and Kaledio Potter

March 16, 2024

Building a people counting solution using computer vision

Date:2nd March, 2024

Author
Olaoye Favour, Kaledio Potter

Abstract:
The development and implementation of people counting solutions using computer vision have gained significant attention in various domains, including retail, transportation, and public spaces. This paper presents an overview of the process involved in building a people counting solution using computer vision techniques. The proposed solution aims to accurately count the number of people in a given area by leveraging the capabilities of computer vision algorithms and machine learning models.

The paper begins by discussing the importance of people counting in different applications, such as crowd management, resource allocation, and customer behavior analysis. It highlights the limitations of traditional manual counting methods and emphasizes the need for automated and reliable solutions.

Next, the paper delves into the technical aspects of building a people counting system. It provides a comprehensive overview of computer vision techniques, including object detection, tracking, and image segmentation, which form the foundation of the proposed solution. The integration of machine learning algorithms, such as convolutional neural networks (CNNs), for accurate people detection and tracking is also explored.

Furthermore, the paper discusses the data acquisition process, including camera selection, placement, and calibration. It addresses challenges such as occlusions, lighting conditions, and camera perspective, which can impact the accuracy of people counting.

The subsequent section covers the implementation of the proposed solution, including the preprocessing of video data, feature extraction, and the training of machine learning models. It also presents different evaluation metrics and techniques to assess the performance of the people counting system.

Additionally, the paper discusses potential deployment scenarios and considerations for scaling the solution to handle large-scale environments. It highlights the importance of real-time processing, scalability, and robustness in practical applications.

Finally, the paper concludes by summarizing the key findings and highlighting future research directions in the field of people counting using computer vision. It emphasizes the potential for

improving accuracy, efficiency, and adaptability of the solution through advancements in deep learning, sensor technologies, and data fusion techniques.

Overall, this paper provides a comprehensive overview of the process involved in building a people counting solution using computer vision. It serves as a valuable resource for researchers, engineers, and practitioners interested in developing accurate and reliable people counting systems for various applications.

Introduction:

In today's world, there is a growing need for accurate and automated people counting solutions in various domains. Whether it is managing crowd flow in retail stores, optimizing resource allocation in transportation hubs, or analyzing customer behavior in public spaces, the ability to count the number of people accurately is crucial. Traditional manual counting methods are often time-consuming, prone to errors, and lack scalability. This has led to the rise of computer vision as a powerful tool to address these challenges.

Computer vision, a subfield of artificial intelligence, focuses on enabling computers to understand and interpret visual information from images or videos. By leveraging advanced algorithms and machine learning models, computer vision has emerged as a promising technology for people counting. In this paper, we will explore the process of building a people counting solution using computer vision techniques.

The objective of this paper is to provide a comprehensive overview of the technical aspects involved in developing a people counting system based on computer vision. We will delve into the algorithms and methodologies used for accurate people detection, tracking, and counting. Additionally, we will discuss the challenges associated with data acquisition, such as camera selection, placement, and calibration, which are critical for achieving reliable results.

The implementation of the proposed solution will be explored, covering key steps such as video data preprocessing, feature extraction, and the training of machine learning models. We will also discuss evaluation metrics and techniques to assess the performance of the people counting system, ensuring its accuracy and reliability.

Furthermore, this paper will address potential deployment scenarios and considerations for scaling the solution to handle large-scale environments. Real-time processing, scalability, and robustness will be emphasized, as they are crucial factors in practical applications.

By the end of this paper, readers will gain a comprehensive understanding of the process involved in building a people counting solution using computer vision. This knowledge will be valuable for researchers, engineers, and practitioners interested in developing accurate and reliable people counting systems for various applications.

Overall, the integration of computer vision techniques into people counting solutions has the potential to revolutionize crowd management, resource allocation, and customer behavior analysis. Through advancements in deep learning, sensor technologies, and data fusion techniques, we anticipate significant improvements in the accuracy, efficiency, and adaptability of people counting systems.

II. Understanding the Problem:

Building a people counting solution using computer vision involves addressing several key challenges and understanding the intricacies of accurately detecting and tracking individuals in various environments. This section aims to provide a comprehensive understanding of the problem at hand.

1. Importance of People Counting:
Accurate people counting is crucial in numerous applications. In retail environments, it helps optimize staffing levels, monitor customer flow, and analyze conversion rates. In transportation hubs, it aids in crowd management, resource allocation, and planning. In public spaces, it enables the analysis of visitor patterns and behavior. Understanding the significance of people counting in these contexts highlights the need for reliable and automated solutions.

2. Limitations of Manual Counting:
Traditional manual counting methods are prone to errors, subjective interpretations, and inefficiencies. They require significant human effort and are not scalable for large areas or real-time analysis. Moreover, manual counting may not be feasible in certain scenarios due to privacy concerns or the need for continuous monitoring. Thus, automated people counting solutions using computer vision offer a more efficient and accurate alternative.

3. Technical Challenges:
Building a people counting solution using computer vision involves overcoming various technical challenges:

a. Object Detection: Accurately detecting individuals in images or video frames is a fundamental component of people counting. It requires robust object detection algorithms capable of identifying people with high accuracy, even in complex scenarios with occlusions, varying scales, and diverse poses.

b. Object Tracking: Once individuals are detected, it is essential to track their movements across frames to maintain accurate counts. Object tracking algorithms must handle challenges such as occlusions, appearance changes, and crowded scenes to ensure reliable tracking.

c. Image Segmentation: In some cases, segmenting the image to distinguish individuals from the background can enhance accuracy. Image segmentation techniques are employed to separate foreground objects, specifically people, from the surrounding environment.

d. Data Acquisition: Choosing the right cameras, their placement, and calibration are critical factors for achieving accurate people counting. Factors such as lighting conditions, camera perspective, and field of view need to be considered to minimize errors and maximize the system's performance.

4. Machine Learning and Deep Learning:
Machine learning, particularly deep learning, plays a pivotal role in developing robust people counting solutions. Convolutional neural networks (CNNs) are commonly utilized for people detection and tracking tasks. Training these models requires annotated datasets and techniques such as transfer learning to adapt pre-trained models to specific counting scenarios.

5. Evaluation Metrics:
Measuring the performance of a people counting system is essential to assess its accuracy and reliability. Evaluation metrics such as precision, recall, F1 score, and intersection over union (IoU) are commonly used to quantify the system's performance against ground truth data or manual counts.

By understanding the challenges and intricacies involved in people counting using computer vision, researchers and practitioners can tailor their approaches to address these issues effectively. The subsequent sections of this paper will delve into the technical details and methodologies employed to build a robust and accurate people counting solution.

III. Data Collection on "Building a People Counting Solution Using Computer Vision":

Data collection is a crucial step in building a people counting solution using computer vision. High-quality and representative data play a vital role in training accurate machine learning models and ensuring reliable performance. In this section, we will discuss the various aspects of data collection for a people counting system.

1. Camera Selection: The choice of camera is an important consideration for data collection. Factors such as resolution, frame rate, field of view, and low-light capabilities should be taken into account. The camera should be capable of capturing clear and detailed images or videos of the area where people counting will be performed.

2. Camera Placement: Proper camera placement is essential to capture the desired field of view and ensure optimal data collection. Factors such as camera height, angle, and location should be carefully determined to minimize occlusions and maximize coverage of the target area. It may be necessary to perform multiple camera placements to cover larger spaces or areas with complex layouts.

3. Calibration: Calibration is the process of mapping the camera's 2D image coordinates to real-world 3D coordinates. This step is crucial for accurate distance estimation and size measurement

of people in the scene. Calibration can be performed using calibration patterns or known reference points in the environment.

4. Annotation: To train machine learning models, labeled data is required. Manual annotation involves labeling each person or object of interest in the collected images or videos. Annotations can include bounding boxes around individuals or pixel-level segmentation masks. Annotation tools and techniques should be chosen to ensure efficiency and accuracy during the annotation process.

5. Diversity of Data: It is important to collect a diverse range of data to account for variations in lighting conditions, camera perspectives, occlusions, and crowd densities. This helps in training a robust model that can handle different scenarios and generalize well to unseen data.

6. Privacy Considerations: When collecting data in public spaces, privacy concerns should be addressed. It is essential to adhere to legal and ethical guidelines regarding the collection and storage of personal information. Privacy measures such as blurring or anonymizing faces can be implemented to protect individuals' identities.

7. Data Preprocessing: Preprocessing steps may be required to enhance the quality of the collected data. This can include noise reduction, image stabilization, contrast adjustment, and normalization. Preprocessing steps help in improving the performance of the people counting system by ensuring cleaner and more consistent input data.

By carefully considering these aspects of data collection, we can acquire the necessary data to train and evaluate a people counting solution using computer vision. Proper camera selection, placement, calibration, annotation, and diversity of data contribute to the development of robust and accurate machine learning models. Adhering to privacy considerations and performing appropriate data preprocessing further enhances the overall performance of the system.

IV. Computer Vision Techniques on "Building a People Counting Solution Using Computer Vision":

Computer vision techniques form the foundation of a people counting solution using computer vision. In this section, we will explore the key techniques employed in detecting, tracking, and counting individuals in images or videos.

1. Object Detection:
Object detection algorithms are used to identify and locate people within an image or video frame. There are various object detection approaches, including:

a. Haar cascades: Haar cascades use a set of pre-defined features and a cascade of classifiers to detect objects. This approach is computationally efficient but may struggle with complex scenarios and occlusions.

   b. Histogram of Oriented Gradients (HOG): HOG analyzes the distribution of gradient orientations in an image to detect objects. It is effective in detecting people based on their shape and appearance, but it may be sensitive to variations in lighting conditions.

   c. Deep Learning-based Object Detection: Convolutional neural networks (CNNs) have revolutionized object detection. Models like Faster R-CNN, YOLO (You Only Look Once), and SSD (Single Shot MultiBox Detector) have shown remarkable performance in detecting people accurately and efficiently.

2. Object Tracking:
Object tracking allows the system to follow the detected individuals across frames, enabling continuous counting. Various tracking algorithms can be employed, including:

   a. Kalman Filters: Kalman filters use a probabilistic model to estimate the state of an object over time. They are commonly used for tracking and can handle occlusions and noisy measurements to some extent.

   b. MeanShift and CamShift: MeanShift and CamShift algorithms use color histograms to track objects based on their appearance. They are effective in scenarios with consistent and distinct color information.

   c. Deep Learning-based Tracking: Deep learning-based trackers utilize CNNs to track objects by learning appearance features and matching them across frames. Trackers like Deep SORT (Simple Online and Realtime Tracking) have shown excellent performance in people tracking tasks.

3. Image Segmentation:
Image segmentation techniques can be employed to separate individuals from the background, which can improve the accuracy of people counting. Some common segmentation methods include:

   a. Semantic Segmentation: Semantic segmentation assigns each pixel in an image to a specific class, such as "person" or "background." It provides pixel-level segmentation masks, enabling precise delineation of individuals.

   b. Instance Segmentation: Instance segmentation goes a step further and distinguishes between different instances of the same class. It provides separate masks for each individual, allowing for accurate counting and tracking.

4. Deep Learning Models:
Deep learning models, particularly CNNs, have shown significant advancements in people counting tasks. Pre-trained CNN models like ResNet, VGG, or MobileNet can be used as feature extractors to capture high-level representations of individuals. These features can be fed into classification or regression models to count the number of people accurately.

5. Post-processing and Filtering:

Post-processing techniques can be applied to refine the results and improve the accuracy of the people counting system. This may involve removing false detections, handling occlusions, and performing data association to match individuals across frames.

By leveraging these computer vision techniques, a people counting solution can accurately detect, track, and count individuals in various scenarios. The integration of deep learning models enhances the system's performance and robustness, enabling it to handle complex and crowded environments effectively.

Implementation of "Building a People Counting Solution Using Computer Vision":

Implementing a people counting solution using computer vision involves several key steps, from data preprocessing to model training and deployment. In this section, we will outline the implementation process for building a people counting system.

1. Data Preprocessing:
   - Load and preprocess the collected data, including images or videos.
   - Perform necessary data cleaning, noise reduction, and image stabilization techniques.
   - Apply calibration parameters to correct for camera distortions and obtain accurate measurements.

2. Object Detection:
   - Utilize a pre-trained object detection model, such as Faster R-CNN or YOLO, to detect people in each frame.
   - Extract bounding box coordinates and confidence scores for each detected person.

3. Object Tracking:
   - Initialize object trackers, such as the Deep SORT algorithm, to track individuals across frames.
   - Update the tracker with the bounding box coordinates of detected people in each frame.
   - Maintain a unique ID for each tracked person to enable accurate counting.

4. Counting and Aggregation:
   - Determine the criteria for counting, such as counting each person once or counting unique individuals over a period of time.
   - Monitor the state of each tracked person and increment the count accordingly.
   - Apply filtering techniques to handle occlusions, false detections, or noise in the tracking results.
   - Aggregate the counts over time to obtain cumulative or real-time people counts.

5. Evaluation and Optimization:
   - Assess the performance of the people counting system using evaluation metrics such as precision, recall, or F1 score.

   - Fine-tune parameters, thresholds, or algorithms based on the evaluation results to improve accuracy and robustness.
   - Perform cross-validation and testing on different datasets to ensure the generalizability of the solution.

6. Deployment:
   - Implement the people counting solution in a real-time or near-real-time setting.
   - Consider the hardware and computational requirements for efficient processing.
   - Integrate the solution into the desired application or environment, such as retail stores, transportation hubs, or public spaces.
   - Ensure the solution meets any specific deployment constraints, such as privacy regulations or real-time processing requirements.

7. Continuous Monitoring and Maintenance:
   - Regularly monitor the performance of the deployed system to detect and address any issues or drift in accuracy.
   - Collect feedback and user data to further optimize the solution and enhance its performance over time.
   - Maintain the system by updating the models, retraining on new data, and adapting to changing environmental conditions.

It is important to note that the specific implementation details may vary depending on the chosen algorithms, libraries, and programming languages. Additionally, integrating the people counting solution with visualization tools or analytics platforms can provide valuable insights and actionable information based on the collected data.

By following these implementation steps and continuously improving the system based on evaluation and feedback, a robust and accurate people counting solution using computer vision can be developed and deployed in various real-world applications.

VI. Performance Evaluation on "Building a People Counting Solution Using Computer Vision":

Performance evaluation is a critical step in assessing the accuracy and reliability of a people counting solution built using computer vision techniques. In this section, we will discuss the key aspects of performance evaluation and the metrics used to evaluate the system's performance.

1. Ground Truth Data:
To evaluate the performance of a people counting system, ground truth data is essential. Ground truth refers to manually annotated data that provides the true count or position of people in a given scene. Ground truth data can be obtained through manual counting, manual annotation of images or videos, or using other reliable counting methods like sensors or manual clickers. Ground truth data serves as a reference to compare the system's output and measure its accuracy.

2. Evaluation Metrics:
Several evaluation metrics can be used to assess the performance of a people counting system:

a. Counting Accuracy: Counting accuracy measures how accurately the system counts the number of people. It is usually calculated as the absolute difference or percentage difference between the system's count and the ground truth count.

b. Precision and Recall: Precision measures the proportion of correctly detected people out of all the detected people by the system. Recall measures the proportion of correctly detected people out of all the ground truth people. High precision and recall indicate accurate detection and counting.

c. F1 Score: The F1 score is the harmonic mean of precision and recall. It provides a single metric that balances both precision and recall. F1 score is a commonly used metric to evaluate the overall performance of a people counting system.

d. Intersection over Union (IoU): IoU measures the overlap between the bounding boxes or segmentation masks generated by the system and the ground truth annotations. It is particularly useful for evaluating object detection and segmentation tasks.

e. Tracking Accuracy: In systems that perform tracking, tracking accuracy metrics such as ID switches, fragmentation rate, and tracking precision can be used. These metrics assess how well the system maintains the identity and trajectory of individuals across frames.

3. Cross-Validation and Testing:
To ensure unbiased evaluation, the collected dataset is often divided into training and testing subsets. Cross-validation techniques, such as k-fold cross-validation, can be employed to evaluate the system's performance on different subsets of the data. Testing the system on unseen data helps assess its generalization capabilities and robustness.

4. Performance Analysis:
In addition to numerical metrics, visual analysis of the system's output is crucial. Examining the system's detections, tracking trajectories, and counting results can provide insights into its strengths and weaknesses. Visual analysis helps identify common failure cases, such as occlusions, complex backgrounds, or lighting conditions, which can guide improvements in the system.

5. Iterative Improvement:
Performance evaluation is an iterative process. By analyzing the system's performance and identifying areas for improvement, adjustments can be made to the algorithms, data collection process, or model architecture. Iteratively refining the system based on performance evaluation ensures progress towards an accurate and reliable people counting solution.

By employing rigorous performance evaluation techniques, researchers and practitioners can quantify the accuracy and reliability of a people counting system. Evaluation metrics, ground truth data, and visual analysis contribute to a comprehensive assessment of the system's performance, enabling iterative improvements and advancements in the solution.

VII. Optimization and Deployment on "Building a People Counting Solution Using Computer Vision":

Optimization and deployment are crucial stages in building a people counting solution using computer vision. In this section, we will discuss the key aspects of optimization and deployment to ensure the solution's efficiency, accuracy, and successful integration into real-world applications.

1. Algorithm Optimization:
Optimizing the algorithms used in the people counting solution can significantly improve its performance. Some optimization techniques include:

a. Model Selection: Experiment with different pre-trained models or architectures to find the most suitable one for the specific application. Consider trade-offs between accuracy and computational efficiency.

b. Hyperparameter Tuning: Adjust hyperparameters of the algorithms, such as learning rates, thresholds, or feature extraction parameters, to optimize their performance on the specific dataset and task.

c. Algorithmic Enhancements: Explore advanced techniques like data augmentation, transfer learning, or ensembling to improve the system's accuracy and robustness.

d. Hardware Acceleration: Utilize hardware acceleration techniques such as GPU (Graphics Processing Unit) or specialized AI chips to speed up processing and inference time.

2. Real-time Processing:
In many applications, real-time or near-real-time processing is crucial. To achieve real-time performance:

a. Efficient Implementation: Optimize the codebase and algorithms for faster execution. Utilize libraries and frameworks that provide hardware acceleration support.

b. Parallel Processing: Exploit parallel processing capabilities, such as multi-threading or distributed computing, to speed up computations.

c. Hardware Considerations: Choose hardware configurations that can handle the computational requirements of the people counting solution efficiently.

3. Integration with Existing Infrastructure:

To deploy the people counting solution seamlessly into an existing infrastructure or application:

a. API or SDK Integration: Develop an API (Application Programming Interface) or SDK (Software Development Kit) to facilitate the integration of the people counting functionality into third-party applications or systems.

b. Compatibility with Frameworks: Ensure compatibility with popular frameworks or platforms commonly used in the target application domain, such as retail analytics software or surveillance systems.

c. Scalability: Design the solution with scalability in mind, allowing it to handle varying workloads and accommodate increased data processing requirements.

4. Deployment Considerations:
When deploying the people counting solution, several factors need to be taken into account:

a. Environmental Factors: Consider factors such as lighting conditions, camera angles, or potential occlusions in the deployment environment. Adapt the solution to handle specific challenges posed by the environment.

b. Privacy and Security: Implement privacy measures to ensure compliance with regulations and protect the privacy of individuals captured by the system. Anonymize or encrypt data as necessary.

c. User Interface and Visualization: Develop a user-friendly interface that allows users to interact with the people counting system and visualize the collected data effectively.

d. Error Handling and Monitoring: Implement robust error handling mechanisms and monitoring systems to identify and address issues quickly. Regularly monitor the system's performance to ensure its reliability.

5. Continuous Improvement:
Deploying a people counting solution is not the end of the development process. Continuously monitor the system's performance, collect user feedback, and analyze the data to identify areas for improvement. Update the solution with new data, retrain models, and integrate advancements in computer vision techniques to enhance accuracy and adaptability.

By optimizing the algorithms, ensuring real-time processing, integrating with existing infrastructure, and considering deployment factors, a people counting solution can be effectively deployed and integrated into various applications. Continuous improvement and monitoring ensure the solution remains accurate, efficient, and in line with the evolving needs of the target environment.

Conclusion:

Building a people counting solution using computer vision is a challenging task that requires a systematic approach and careful consideration of various factors. In this guide, we have explored the key steps involved in developing such a solution, including data preprocessing, object detection, object tracking, counting, evaluation, optimization, and deployment.

By leveraging computer vision techniques, such as object detection and tracking algorithms, accurate and real-time people counting can be achieved. The implementation process involves data preprocessing to ensure data quality, applying object detection models to detect people, employing object tracking algorithms to maintain individual identities across frames, and performing counting and aggregation to obtain accurate counts.

Performance evaluation is a crucial step in assessing the accuracy and reliability of the people counting system. Metrics such as counting accuracy, precision, recall, F1 score, and intersection over union (IoU) provide quantitative measures of the system's performance. Ground truth data and visual analysis help validate the system's outputs and identify areas for improvement.

Optimization and deployment are essential for ensuring the efficiency and successful integration of the solution. Algorithm optimization, real-time processing, integration with existing infrastructure, and considering deployment factors such as environmental conditions, privacy, and security are key considerations. Continuous monitoring and improvement are necessary to maintain the system's accuracy, adaptability, and performance over time.

Building a people counting solution using computer vision has various applications, including retail analytics, transportation management, and crowd monitoring. Accurate people counting provides valuable insights for resource allocation, crowd control, and business decision-making.

As computer vision techniques continue to advance, incorporating machine learning and deep learning models, the accuracy and reliability of people counting solutions are expected to improve further. With the right approach and continuous improvement, a robust people counting solution can be developed and deployed, contributing to more efficient operations and enhanced user experiences in various domains.

References

1. Jian, Yanan, Fuxun Yu, Simranjit Singh, and Dimitrios Stamoulis. "Stable Diffusion For Aerial Object Detection." *arXiv preprint arXiv:2311.12345* (2023).

2. Lapid, R., Haramaty, Z., & Sipper, M. (2022, October 31). An Evolutionary, Gradient-Free, Query-Efficient, Black-Box Algorithm for Generating Adversarial Instances in Deep Convolutional Neural Networks. *Algorithms*, *15*(11), 407. https://doi.org/10.3390/a15110407
3. Li, C., Wang, H., Zhang, J., Yao, W., & Jiang, T. (2022, October). An Approximated Gradient Sign Method Using Differential Evolution for Black-Box Adversarial Attack. *IEEE Transactions on Evolutionary Computation*, *26*(5), 976–990. https://doi.org/10.1109/tevc.2022.3151373
4. Chen, J., Huang, G., Zheng, H., Zhang, D., & Lin, X. (2023, October). Graphfool: Targeted Label Adversarial Attack on Graph Embedding. *IEEE Transactions on Computational Social Systems*, *10*(5), 2523–2535. https://doi.org/10.1109/tcss.2022.3182550
5. Wang, J., Shi, L., Zhao, Y., Zhang, H., & Szczerbicki, E. (2022, October 26). Adversarial attack algorithm for traffic sign recognition. *Multimedia Tools and Applications*. https://doi.org/10.1007/s11042-022-14067-5
6. Liu, H., Xu, Z., Zhang, X., Xu, X., Zhang, F., Ma, F., Chen, H., Yu, H., & Zhang, X. (2023, June 26). SSPAttack: A Simple and Sweet Paradigm for Black-Box Hard-Label Textual Adversarial Attack. *Proceedings of the AAAI Conference on Artificial Intelligence*, *37*(11), 13228–13235. https://doi.org/10.1609/aaai.v37i11.26553
7. Sawant, A., & Giallanza, T. (2022, August 27). ZQBA: A Zero-Query, Boosted Ambush Adversarial Attack on Image Retrieval. *International Journal on Cybernetics & Informatics*, *11*(4), 53–65. https://doi.org/10.5121/ijci.2022.110404
8. Xu, G., Shao, H., Cui, J., Bai, H., Li, J., Bai, G., Liu, S., Meng, W., & Zheng, X. (2023, September). GenDroid: A query-efficient black-box android adversarial attack framework. *Computers & Security*, *132*, 103359. https://doi.org/10.1016/j.cose.2023.103359
9. Jaiswal, Ayush, Simranjit Singh, Yue Wu, Pradeep Natarajan, and Premkumar Natarajan. "Keypoints-aware object detection." In *NeurIPS 2020 Workshop on Pre-registration in Machine Learning*, pp. 62-72. PMLR, 2021.
10. Bai, Y., Wang, Y., Zeng, Y., Jiang, Y., & Xia, S. T. (2023, January). Query efficient black-box adversarial attack on deep neural networks. *Pattern Recognition*, *133*, 109037. https://doi.org/10.1016/j.patcog.2022.109037
11. Dong, H., Dong, J., Wan, S., Yuan, S., & Guan, Z. (2023, December). Transferable adversarial distribution learning: Query-efficient adversarial attack against large language models. *Computers & Security*, *135*, 103482. https://doi.org/10.1016/j.cose.2023.103482
12. Peng, H., Guo, S., Zhao, D., Zhang, X., Han, J., Ji, S., Yang, X., & Zhong, M. (2023). TextCheater: A Query-Efficient Textual Adversarial Attack in the Hard-Label Setting. *IEEE Transactions on Dependable and Secure Computing*, 1–16. https://doi.org/10.1109/tdsc.2023.3339802

13. Cheng, Minhao, Simranjit Singh, Patrick Chen, Pin-Yu Chen, Sijia Liu, and Cho-Jui Hsieh. "Sign-opt: A query-efficient hard-label adversarial attack." *arXiv preprint arXiv:1909.10773* (2019).