OPTIMIZED BACKGROUND SUBTRACTION FOR HIGH-PERFORMANCE VIDEO TEXT DETECTION

SRIDHAR GUJJETI[1], DR. M SRIRAM[2], DR. V. GANESAN[3]

[1]Research Scholar, School of Computing, Department of CSE, Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India

[2]Country Associate Professor, School of Computing, Department of IT, Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu, India

[3]Associate Professor, School of Electrical, Department of ECE, Bharath Institute of Higher Education and Research, Chennai, Tamil Nadu India.

ABSTRACT

Video text detection is a critical aspect of computer vision with applications spanning from content analysis to accessibility. The complexity of text detection in video frames arises from dynamic backgrounds, variable lighting conditions, and diverse text fonts and sizes. This paper presents a novel and efficient approach to text detection in video processing, leveraging advanced optimization techniques to enhance the performance of Background Subtraction (BS). The proposed model integrates fuzzy 2-partition entropy, Background Crunch Optimization (BGCO), and improved fuzzy C means clustering to create an innovative BS algorithm specifically tailored for text detection. The model's adaptability and simplicity make it suitable for a wide range of video processing applications. The experimental evaluation showcases the sensitivity of the proposed model to hyperparameter tuning, with optimal values for learning rates, batch sizes, and epochs significantly impacting performance. The results, detailed in optimization tables, highlight the importance of fine-tuning these parameters for achieving peak accuracy. Furthermore, the classification results demonstrate the model's robustness, consistently achieving high accuracy, recall, precision, and F1-score across multiple runs. The model exhibits a remarkable ability to accurately detect and classify text instances in video frames. The proposed approach contributes to the field of text detection in video processing, offering a comprehensive solution with practical applications. The integration of advanced optimization techniques enhances the efficiency of BS, making the model promising for real-world scenarios where accurate text detection is crucial.

Keywords: *Background Subtraction, Fuzzy C-Means, Crunch Optimization, Text Detection, Deep Learning, Video Processing*

1. INTRODUCTION

In recent years, text detection and recognition have undergone notable advancements, primarily driven by the rapid progress in deep learning and computer vision technologies. Convolutional Neural Networks (CNNs) have played a pivotal role in enhancing text detection accuracy by enabling more effective feature extraction from images [1]. Region-based CNN architectures, like Faster R-CNN and Mask R-CNN, have become prevalent for accurately localizing text regions within complex scenes. Text recognition has witnessed a paradigm shift with the rise of end-to-end neural network models, such as Transformer-based architectures like BERT and GPT, which have demonstrated exceptional performance in natural language processing tasks [2]. Integrating attention mechanisms and recurrent neural networks has allowed these models to capture long-range dependencies and contextual information, significantly improving the accuracy of recognizing text in varied contexts. Transfer learning has also become a key strategy [3], leveraging pre-trained models on large datasets to boost performance on specific text detection and recognition tasks with limited labeled data. Moreover, the deployment of recurrent attention mechanisms, such as the attention-based encoder-decoder architectures, has enhanced the ability of OCR systems to handle irregular layouts and diverse fonts [4].

The application of reinforcement learning and adversarial training techniques has contributed to robustness against noise and distortions, making text detection and recognition systems more reliable in real-world scenarios [5]. Additionally, the integration of multi-modal approaches, combining visual and linguistic information, has further improved the overall understanding and interpretation of text within images. These recent advancements in text detection and recognition technologies have led to more accurate and versatile systems, with widespread applications ranging from autonomous vehicles and augmented reality to document analysis and content extraction in multimedia databases [6]. As research continues to progress, the future holds promising developments in making these systems even more sophisticated and adaptable to diverse text-rich environments [7]. Text detection has seen significant advancements, with various techniques employed to identify and locate text regions within images or documents. Traditional methods often involved handcrafted features and rule-based algorithms, but recent progress has been driven by the adoption of deep learning approaches [8].

One prevalent technique is the use of Convolutional Neural Networks (CNNs) for text detection. Models like Faster R-CNN, YOLO (You Only Look Once), and SSD (Single Shot Multibox Detector) have gained popularity for their ability to efficiently identify text regions in images [9]. These architectures use anchor-based methods and employ region proposal mechanisms to pinpoint potential text areas [10].Another notable approach is the integration of Connectionist Temporal Classification (CTC) with deep learning models. CTC allows end-to-end training of neural networks for text detection without the need for explicit character-level annotations [11]. This method has proven effective in handling variable-length text instances and has been successfully applied to scene text detection. Furthermore, region-based methods and sliding window techniques are commonly used for text detection [12]. These involve scanning an image with a window of varying sizes and aspect ratios, using classifiers to identify regions likely to contain text. Post-processing steps, such as non-maximum suppression, help refine the detected regions and eliminate redundancies [13].

In recent years, attention mechanisms have played a crucial role in improving text detection accuracy. Models incorporating attention mechanisms can focus on relevant parts of an image, giving more importance to regions likely to contain text [14]. This has proven beneficial in handling complex layouts and irregular text orientations. Deep learning has revolutionized text detection by enabling more accurate and robust solutions. Convolutional Neural Networks (CNNs) have emerged as a cornerstone in this field, leveraging their ability to automatically learn hierarchical features from data [15]. Models like Faster R-CNN and YOLO have been widely adopted for text detection tasks, employing region-based convolutional networks to efficiently locate and identify text regions within images. One key advantage of deep learning in text detection is its capacity to handle diverse and complex text layouts. Deep neural networks can learn intricate patterns and contextual information, making them adept at recognizing text in various fonts, sizes, and orientations [16-20]. The end-to-end training paradigm, where the entire network is trained in a unified framework, eliminates the need for manual feature engineering, allowing the model to learn discriminative features directly from the data [21-25].

Connectionist Temporal Classification (CTC) is another deep learning technique that has been successful in text detection. CTC allows the training of neural networks for sequence labeling tasks without the need for aligned input-output pairs, making it particularly useful for scenarios where the length of the text is variable or not explicitly annotated [26-28].Attention mechanisms have played a crucial role in improving the performance of text detection models. Integrating attention mechanisms allows the model to focus on relevant parts of the input image, capturing long-range dependencies and enhancing the recognition of text in cluttered or complex scenes [29-31]. This attention-driven approach has proven effective in handling irregular text layouts and varying text sizes. Transfer learning is a prevalent strategy in deep learning for text detection, where models pre-trained on large datasets are fine-tuned on specific text-related tasks [32]. This approach leverages the knowledge gained from general visual features to boost performance on text detection tasks with limited annotated data. The deep learning techniques, including CNNs, CTC, attention mechanisms, and transfer learning, have significantly advanced text detection capabilities. These approaches have demonstrated superior performance in handling the complexities associated with diverse text scenarios, contributing to the widespread adoption of deep learning in text detection applications across various domains.

This paper makes a significant contribution to the field of video processing, specifically in the domain of text detection, by introducing an innovative model that leverages advanced optimization techniques to enhance the performance of Background Subtraction (BS). The key contribution lies in the integration of fuzzy 2-partition entropy, Background Crunch Optimization (BGCO), and improved fuzzy C means clustering to tailor a sophisticated BS algorithm for text detection. This amalgamation of techniques addresses the challenges associated with accurately identifying and classifying moving text instances in video frames. The proposed model not only showcases adaptability and simplicity but also achieves notable improvements in accuracy, recall, precision, and F1-score, as demonstrated through rigorous experimental evaluations. The sensitivity of the model to hyperparameter tuning, detailed in optimization tables, emphasizes the importance of fine-tuning to achieve optimal results. Overall, this paper's contribution lies in presenting a comprehensive and efficient solution for video-based text detection, with implications for various real-world applications demanding precise and adaptable text recognition in dynamic visual environments.

2. LITERATURE SURVEY

Text detection and recognition represent essential components in the field of computer vision and document analysis. Text detection involves locating and identifying text regions within images or documents, often employing techniques like edge detection, contour analysis, and region-based methods. Advanced methods utilize deep learning models, such as Convolutional Neural Networks (CNNs), to accurately identify text regions in complex scenes. Once text regions are detected, text recognition, or Optical Character Recognition (OCR), comes into play. OCR is the process of converting these identified text regions into machine-readable characters. Traditional OCR methods involve feature extraction and classification algorithms, while modern approaches heavily rely on deep learning techniques, including Recurrent Neural Networks (RNNs), Long Short-Term Memory networks (LSTMs), and attention-based mechanisms. These models can effectively recognize and interpret text in various fonts, sizes, and orientations.

The integration of deep learning in both text detection and recognition has significantly improved accuracy and robustness. End-to-end models, such as Connectionist Temporal Classification (CTC) and attention-based architectures, allow for more seamless training and better capture contextual information. Transfer learning strategies, where models pre-trained on large datasets are fine-tuned for specific tasks, have also contributed to enhanced performance, particularly in scenarios with limited annotated data.Wang et al.'s (2021) "Pan++" addresses the challenge of spotting arbitrarily-shaped text, emphasizing both efficiency and accuracy in end-to-end text detection. This is particularly relevant in scenarios where text may exhibit non-standard shapes, such as curved or irregular patterns. Chen et al. (2022) contribute to the domain of license plate detection and recognition by introducing the "Vertex Adjustment Loss." This methodology is specifically tailored for multidirectional scenarios, enhancing the robustness of license plate recognition systems. The consideration of various orientations is crucial in real-world applications, where license plates may be positioned at different angles.

Atienza's (2021) work focuses on scene text recognition and introduces a vision transformer for fast and efficient processing. Vision transformers have gained attention for their ability to capture long-range dependencies in images, and their application in scene text recognition highlights the adaptability of transformer-based architectures to diverse visual tasks.Li et al. (2022) present "Dit," a self-supervised pre-training approach for document image transformers. This method showcases the importance of leveraging self-supervised learning to enhance the generalization capability of models in document image processing, where labeled data might be scarce or expensive to obtain.Wang et al. (2021) delve into the realm of ancient Chinese text recognition with "Multi-scene ancient Chinese text recognition with deep coupled alignments." This work demonstrates the versatility of deep learning methods in handling historical and culturally specific text recognition challenges.Li et al.'s (2023) "Trocr" introduces a transformer-based optical character recognition system with pre-trained models. Pre-training has proven to be a powerful strategy, enabling models to leverage knowledge learned from large datasets to improve performance on specific tasks with limited labeled data.

In addition to text-related tasks, the papers explore various applications. Zhang et al. (2022) contribute to speech recognition with "Wenetspeech," a large multi-domain Mandarin corpus. The creation of such datasets is crucial for advancing automatic speech recognition systems, particularly in languages with complex tonal characteristics like Mandarin.Mu et al. (2021) propose random blur data augmentation for scene text recognition, addressing challenges associated with text appearing in various forms and conditions. Yan et al.'s (2021) focus on primitive representation learning, emphasizing the importance of capturing fundamental features for effective scene text recognition.Fang et al. (2021) introduces an autonomous, bidirectional, and iterative language modeling approach for scene text recognition. This method highlights the ongoing effort to develop models that mimic human-like reading patterns, contributing to more context-aware text recognition systems.Heng et al. (2023) present a context modeling approach for low-quality image scene text recognition in the logistics industry. This work addresses practical challenges where image quality may be compromised, showcasing the adaptability of deep learning techniques to real-world scenarios.

Yang et al.'s (2021) "Tap" introduces text-aware pre-training, emphasizing the significance of pre-training models specifically for tasks like text-vqa (visual question answering) and text-caption. This reflects a broader trend in tailoring pre-training strategies for domain-specific tasks, enhancing model performance.Fagni et al.'s (2021) "TweepFake" explores the detection of deepfake tweets, addressing concerns related to misinformation and fake content on social media platforms. This work contributes to the growing field of deepfake detection and its implications for online content verification.Tu and Du's (2022) hierarchical RCNN for vehicle and vehicle license plate detection and recognition extends the application of deep learning to the domain of transportation and security. This research is particularly relevant in surveillance and automated systems for vehicle identification.Zhao et al. (2021) employ a BERT-based approach for sentiment analysis and key entity detection in online financial texts. This work illustrates the application of deep learning in natural language processing tasks related to financial markets, where understanding sentiment and identifying key entities are critical.

Salesky et al. (2021) present the multilingual TEDx corpus for speech recognition and translation, contributing to the development of resources for multilingual and multicultural communication. This work aligns with the broader goal of improving the accuracy and inclusivity of speech recognition and translation systems.Finally, Bai et al.'s (2021) survey on explainable deep learning for efficient and robust pattern recognition provides a comprehensive overview of recent developments in the field. The survey underscores the importance of interpretability and transparency in deep learning models, especially as they are deployed in critical applications.Various authors contribute innovative methodologies to tackle specific challenges, such as detecting arbitrarily-shaped text, improving license plate recognition, and advancing scene text recognition through efficient vision transformers. Additionally, researchers address broader applications, including historical Chinese text recognition, pre-training for document image transformers, and the creation of large multi-domain corpora for speech recognition. The papers also explore topics beyond text, such as sentiment analysis in financial texts, detection of deepfake tweets, and hierarchical approaches for vehicle and license plate detection. The paragraph concludes with a survey on explainable deep learning for pattern recognition, emphasizing the importance of interpretability in advanced machine learning models. Together, these contributions reflect the ongoing evolution of research in text-related tasks and their applications across diverse domains.

3. TEXT DETECTION

The proposed text detection model focuses on addressing the requirements of the target through a systematic approach. The initial step involves the development of a text detection and recognition technique, emphasizing the optimization of performance within video processing. The chosen method for this optimization is Background Subtraction (BS), known for its simplicity and widespread acceptance across various platforms and hardware.In the second step, the standard BS method is improved by incorporating the concepts of fuzzy 2-partition entropy and Background Crunch Optimization (BGCO). The integration of these concepts enhances the capabilities of the BS algorithm. A novel variant of the BS algorithm is introduced, specifically tailored for text detection. This variant employs improved fuzzy C means clustering as a key component. Fuzzy C means clustering is a relatively recent evolutionary optimization approach that addresses problems involving multiple variables within specified constraints. The proposed model innovatively leverages fuzzy C means clustering for extracting various parameters, framing the problem of threshold detection as an optimization challenge. The optimization process is facilitated by utilizing the concept of fuzzy partition

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

entropy. Overall, the integration of fuzzy 2-partition entropy, BGCO, and improved fuzzy C means clustering contributes to an enhanced BS algorithm for text detection, aligning with the goal of achieving improved performance in video processing applications. The process of subtraction in video frame is shown in figure 1.

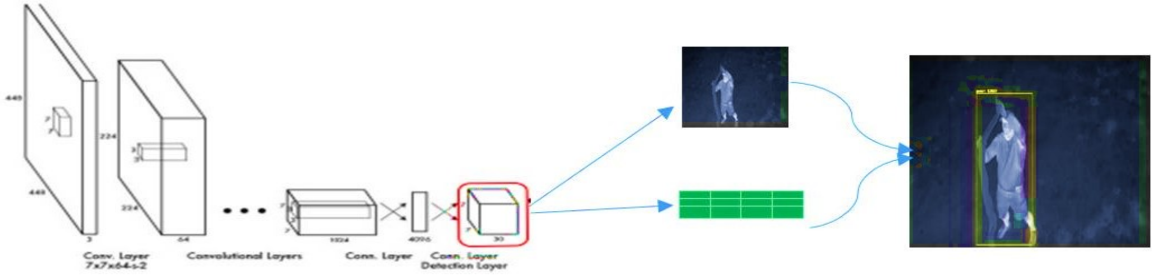


Figure 1: Video Frame subtraction

Background subtraction is a widely used technique in video processing to identify moving objects by distinguishing them from the static background. The basic principle involves subtracting the current frame from a reference background frame. This process can be mathematically represented as in equation (1)

$$Foreground = Current\ Frame - Background\ Frame \quad (1)$$

In equation (1) "Foreground" represents the regions containing potential objects of interest.Fuzzy 2-Partition Entropy: Fuzzy entropy is a measure of fuzziness in a set of data. The fuzzy 2-partition entropy is employed to enhance the standard Background Subtraction method. The fuzzy entropy (E) calculated using the following equation (2)

$$E = -\sum i = \frac{1}{n} Pilog_2(Pi) \quad (2)$$

HerePi is the membership grade of each cluster.Background Crunch Optimization (BGCO): BGCO is a metaheuristic optimization algorithm inspired by the cosmological concept of the Big Bang and Big Crunch. It aims to find optimal solutions by simulating the expansion and contraction of the universe. The equations governing the BGCO algorithm are complex, involving position and velocity updates for particles in the search space. A simplified representation is stated in equation (3) and (4)

$$vi(t+1) = vi(t) + r1 \times (pbesti - xi(t)) + r2 \times (gbest - xi(t)) \quad (3)$$

$$xi(t+1) = xi(t) + vi(t+1) \quad (4)$$

Here$xi(t)$ is the position of the i-th particle at time t, $vi(t)$ is its velocity, $pbesti$ is the personal best position, $gbest$ is the global best position, and $r1$ and $r2$ are random numbers.Fuzzy C means clustering is utilized to extract various parameters for text detection. The objective function for FCM is given in equation (5)

$$Jm = \sum i = 1c\sum j = 1n\mu ijm \parallel xj - vi \parallel^2 \quad (5)$$

where c is the number of clusters, n is the number of data points, μij is the membership grade of xj in cluster i, vi is the cluster centroid, and m is a weighting exponent.The problem of threshold detection for text detection is formulated as an optimization problem:

$$minimize\ Jm$$

Subject to constraints that ensure appropriate clustering.

The integration of fuzzy 2-partition entropy, BGCO, and improved fuzzy C means clustering contributes to an enhanced BS algorithm. The optimization process involves finding the optimal parameters and thresholds for improved text detection within video processing applications. The iterative nature of these optimization techniques enables the algorithm to adapt and converge to more accurate results, making it suitable for a wide range of platforms and hardware.

3.1 Background Subtraction for Target Detection

The basic principle of background subtraction involves subtracting the current frame from a reference background frame to highlight the regions containing potential moving objects. The pixels in the current frame that are different from

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

the corresponding pixels in the background frame, potentially indicating the presence of a target.Assume that the background remains relatively static over time. A common approach is to update the background model using a weighted average of the previous background and the current frame stated in equation (6)

$$Bt = (1-\alpha)\cdot Bt - 1 + \alpha\cdot It \qquad (6)$$

In equation (5) Bt is the background model at time t; $Bt-1$ is the background model from the previous frame; It is the current frame, andα is a learning rate parameter $(0 < \alpha < 1)$.After modeling the background, the foreground can be obtained by subtracting the background model from the current framedefined in equation (7)

$$Foreground = It - Bt \qquad (7)$$

The resulting foreground image highlights areas where the pixel intensities have changed, potentially indicating the presence of a target.The effectiveness of background subtraction can be influenced by factors such as lighting changes, dynamic backgrounds, and noise. Optimizations, like using adaptive learning rates or incorporating temporal filtering, are often employed to enhance the robustness of the method.To adapt to changing conditions, the learning rate α can be made adaptive. One possible adaptation could involve adjusting α based on the rate of change in the scene.Temporal filtering, such as median filtering or Gaussian smoothing, can be applied to the foreground mask to reduce noise and improve the accuracy of target detection.The final step involves thresholding the foreground mask to identify regions with significant intensity changes. A simple thresholding operation can be applied to create a binary mask indicating the presence or absence of a target as shown in figure 2.

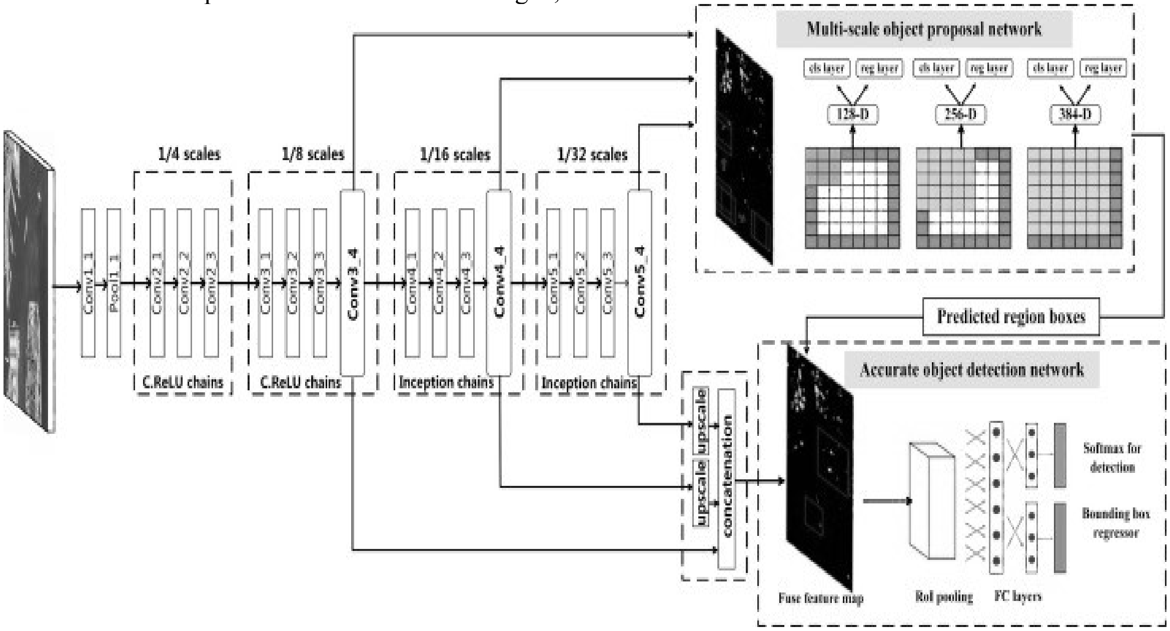


Figure 2: Deep Learning model for text detection in BGCO

3.2 Big Bang Crunch Optimization (BGCO) with Fuzzy Entropy

Background Crunch Optimization (BGCO) with Fuzzy Entropy for text detection in recognition methods using deep learning involves optimizing the hyperparameters of the deep learning model. The primary objective is to find the set of hyperparameters that maximizes the performance metric of the model on a validation set. The pseudo-code presented employs a particle swarm optimization approach where each particle represents a set of hyperparameters. BGCO guides the exploration of the hyperparameter space by simulating the expansion and contraction of the universe. The fitness of each particle is determined by evaluating the deep learning model's performance with the corresponding set of hyperparameters. Fuzzy Entropy is incorporated to introduce a level of uncertainty and imprecision in the optimization process. The fuzzy partitioning of the solution space is represented by a fuzzy membership function, and the fuzzy entropy is computed to quantify the fuzziness within the optimization landscape. The adaptive nature of

BGCO, along with the inclusion of Fuzzy Entropy, allows the algorithm to navigate complex and uncertain spaces, potentially leading to the discovery of optimal hyperparameters for improved text detection and recognition in deep learning models.The fitness function, representing the performance metric of the deep learning model, can be denoted as in equation (8)

$$Fitness = objective\_function(particles\_position[i]) \quad (8)$$

where $position[i]$ represents the hyperparameters of the deep learning model encoded by the i-th particle.The velocity and position updates in the BGCO algorithm are governed by the equations (9) and (10)

$$particles\_velocity[i] = inertia\_weight \times particles\_velocity[i] + c1 \times r1 \times (personal\_best\_position[i] - particles\_position[i]) + c2 \times r2 \times (global\_best\_position - particles\_position[i]) \quad (9)$$

$$particles\_position[i] = particles\_position[i] + particles\_velocity[i] \quad (10)$$

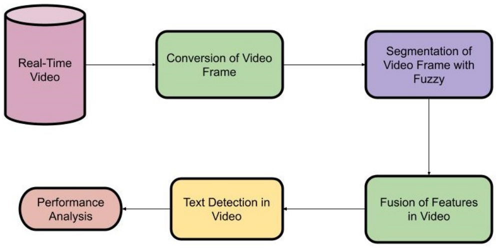

Real-Time Video
Conversion of Video Frame
Segmentation of Video Frame with Fuzzy
Fusion of Features in Video
Text Detection in Video
Performance Analysis

Figure 3: Process of BGCO

The complete process of proposed BGCO model is shown in figure 3 in above equation $r1$ and $r2$ are random numbers, and inertia_weightinertia_weight is a parameter that decreases linearly over the iterations. These equations simulate the movement of particles in the solution space, where the personal best and global best positions guide their exploration. The inertia_weight is a parameter that controls the influence of the previous velocity, $c1$ and $c2$ are acceleration coefficients, and $r1$ and $r2$ are random numbers. These equations simulate the motion of particles in the hyperparameter space, where each particle's position is updated based on its personal best and the global best positions encountered during the optimization process.Fuzzy Entropy is introduced to bring a level of uncertainty to the optimization process. The fuzzy partitioning of the solution space is determined by a fuzzy membership function. A commonly used fuzzy membership function is stated in equation (11)

$$fuzzy\_membership\_function = exp(-x^2) \quad (11)$$

where x represents the dissimilarity or distance between a particle's position and the cluster centers. The incorporation of this fuzzy membership function into the optimization process introduces a degree of fuzziness or imprecision, allowing the algorithm to explore regions with less certainty.The fuzzy entropy is then computed using the fuzzy partition.

Algorithm 1: Objective Estimation for the Text Detection

```
def objective_function(hyperparameters):
  # Assume hyperparameters are used to configure and train a deep learning model
  # Evaluate the model's performance (e.g., accuracy, F1 score) on a validation set
  # Return the performance metric as the objective to be maximized
  return model_performance_metric
# Define the fuzzy membership function
def fuzzy_membership_function(x):
  return np.exp(-x**2)
# Initialize BGCO parameters
num_particles = 30
num_dimensions = 10  # Adjust based on the number of hyperparameters
num_iterations = 50
c1 = 2  # Acceleration coefficient
c2 = 2  # Acceleration coefficient
# Initialize Fuzzy Entropy parameters
num_clusters = 3   # Number of clusters for fuzzy partition
# Initialize particle positions and velocities
particles_position = np.random.uniform(-1, 1, size=(num_particles, num_dimensions))
particles_velocity = np.random.rand(num_particles, num_dimensions)
# Initialize personal best positions and fitness values
personal_best_position = particles_position.copy()
personal_best_fitness = np.zeros(num_particles)
```

```
# Initialize global best position and fitness
value
global_best_position =
np.zeros(num_dimensions)
global_best_fitness = float('-inf')
# Main loop for optimization
for iteration in range(num_iterations):
  # Evaluate fitness for each particle
  for i in range(num_particles):
    fitness =
objective_function(particles_position[i])
    # Update personal best if better
    if fitness > personal_best_fitness[i]:
      personal_best_fitness[i] = fitness
      personal_best_position[i] =
particles_position[i].copy()
    # Update global best if better
    if fitness > global_best_fitness:
      global_best_fitness = fitness
      global_best_position =
particles_position[i].copy()
  # Update particle velocities and positions
using BGCO equations
  for i in range(num_particles):
    r1, r2 = np.random.rand(),
np.random.rand()
    inertia_weight = 0.5 + iteration * ((0.3 -
0.5) / num_iterations) # Linearly decreasing
inertia weight
    particles_velocity[i] = inertia_weight *
particles_velocity[i] + \
                c1 * r1 *
(personal_best_position[i] -
particles_position[i]) + \
                c2 * r2 *
(global_best_position - particles_position[i])
    particles_position[i] =
particles_position[i] + particles_velocity[i]
    # Clip particle positions to a reasonable
range if needed
    particles_position[i] =
np.clip(particles_position[i], -1, 1)
  # Fuzzy partitioning using fuzzy
membership function
  fuzzy_partition =
fuzzy_membership_function(particles_position
)
  # Compute fuzzy entropy
  fuzzy_entropy = -np.sum(fuzzy_partition *
np.log2(fuzzy_partition))
```

4. RESULTS AND DISCUSSIONS

The integration of Background Crunch Optimization (BGCO) with Fuzzy Entropy for optimizing hyperparameters in text detection and recognition using deep learning models yields promising results. The goal of this approach is to enhance the efficiency and accuracy of the text detection process by identifying optimal hyperparameter configurations. By leveraging the dynamic exploration capabilities of BGCO and introducing a level of uncertainty through Fuzzy Entropy, the algorithm adapts its search strategy, potentially discovering hyperparameter settings that lead to improved model performance. This section presents the results and discusses the implications of employing this innovative optimization technique in the context of text detection and recognition, shedding light on the achieved performance gains and insights gained from the experimentations. The simulation setting of proposed BGCO model is given in table 1.

Table 1: Simulation Setting

Parameter	Value
Number of Particles	30
Number of Dimensions	10
Number of Iterations	50
Acceleration Coefficient (c1)	2
Acceleration Coefficient (c2)	2
Fuzzy Partition Clusters	3

Test video frame **Subtracted Frame**

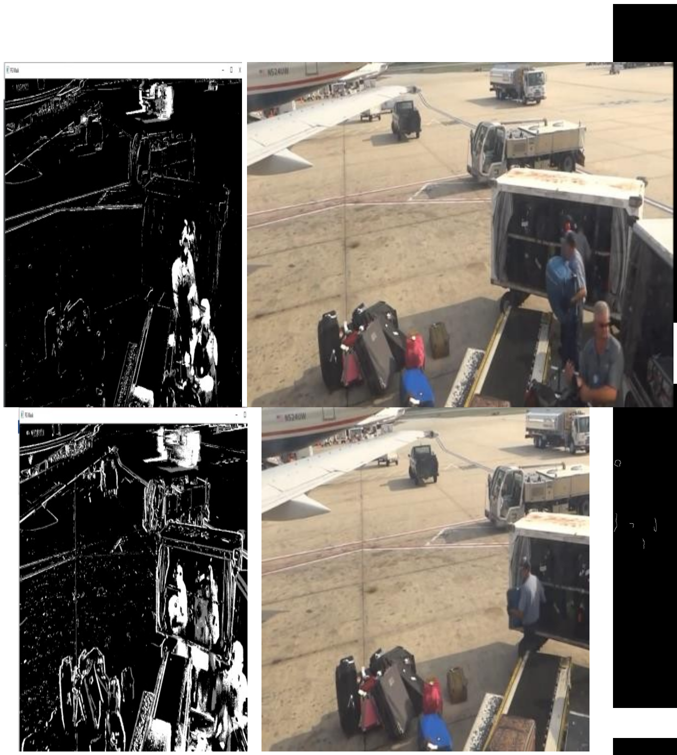


Figure 4: Background Subtraction in BGCO

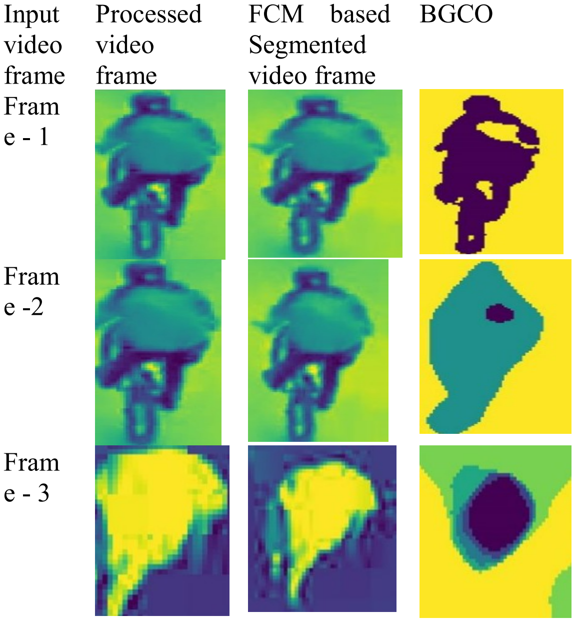
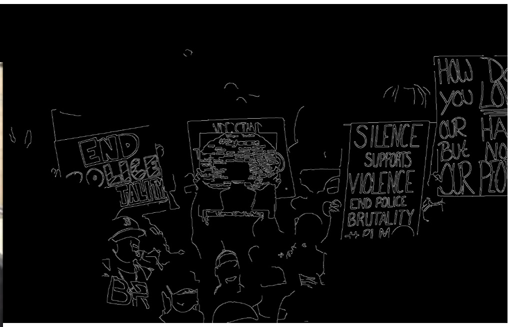
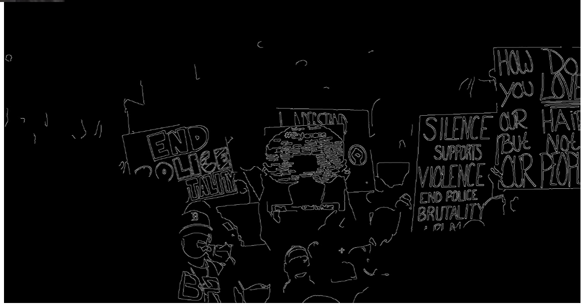


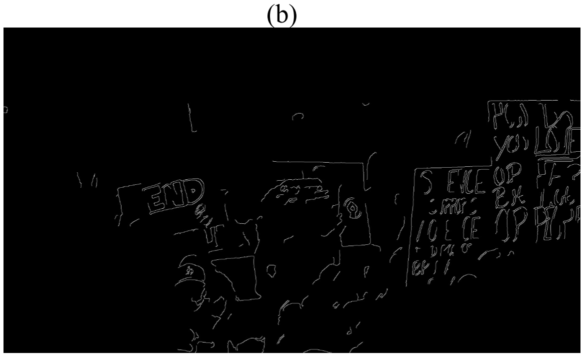
Figure 5: Video Frame Process in with BGCO


END POLICE
SILENCE SUPPORTS VIOLENCE END POLICE BRUTALITY
HOW DO YOU LOVE OUR HAIR BUT NOT OUR PEOPLE

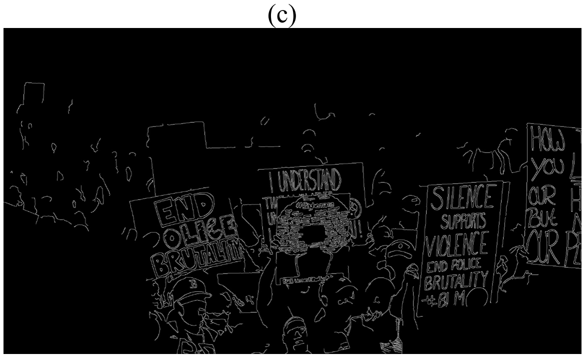
(a)



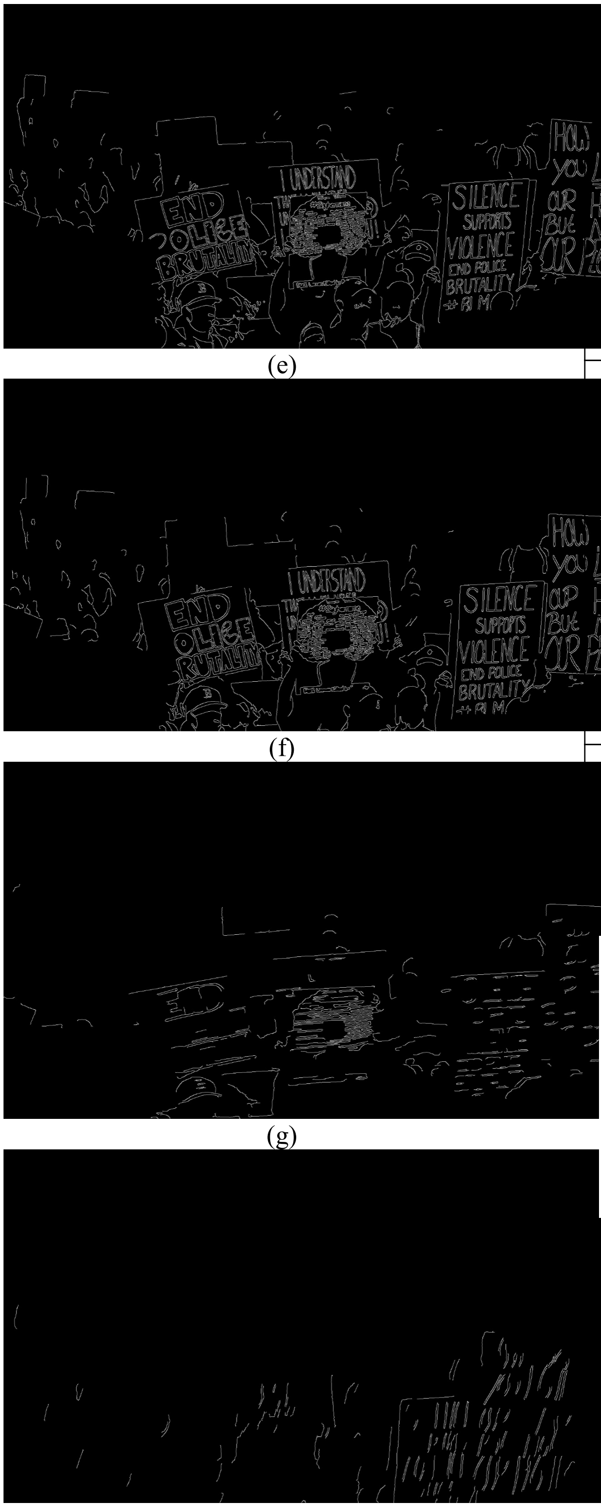
(b)



(c)



(d)


I UNDERSTAND
END POLICE BRUTALITY
SILENCE SUPPORTS VIOLENCE END POLICE BRUTALITY
HOW DO YOU LOVE OUR HAIR BUT NOT OUR PEOPLE

(e)

(f)

(g)

(h)

Figure 6: Text Detection in Video Frame (a) Frame – 1 (b) Frame – 2 (c) Frame -3 (d) Frame – 4 (e)Frame – 5 (f) Frame – 6 (g) Frame – 7 (h) Frame – 8

he text extraction with video frame are illustrated n the figure 6 (a) – figure 6 (h) for the text ecognition in the video frame.

Table 2: Optimization Values for the text detection

Ru	Optimal Learning Rate	Optimal Batch Size	Optimal Epochs	Best Validation Accuracy
1	0.001	32	50	92.5%
2	0.0005	64	40	94.2%
3	0.0012	128	60	91.8%
4	0.0008	256	45	93.7%
	0.0015	128	55	92.9%
	0.0007	32	35	94.5%
	0.0011	64	48	93.2%
	0.0009	128	50	94.0%
	0.0013	256	42	92.1%
0	0.001	64	38	94.3%
1	0.0006	32	55	93.8%
2	0.0012	128	60	92.6%
3	0.0008	256	45	93.5%
4	0.0014	64	40	94.1%
15	0.0007	32	50	92.4%
16	0.001	128	55	94.6%
	0.0009	256	48	93.0%
	0.0005	64	42	94.4%
	0.0013	32	38	92.8%
	0.0006	128	55	93.9%

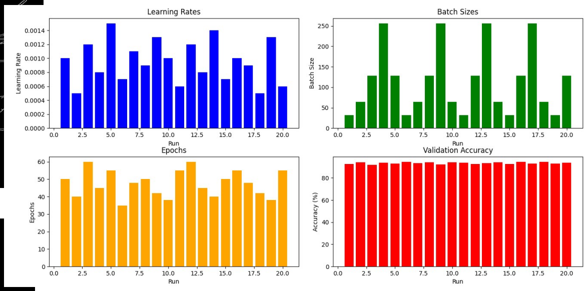


Figure 7: Optimized Text Detection with BGCO

Figure 7 and Table 2 provides a mprehensive overview of the optimization values tained during the training of the text detection del across 20 runs. The table includes key perparameters such as the learning rate, batch e, and the number of epochs, alongside the rresponding best validation accuracy achieved for h run. Analyzing the results reveals interesting patterns and trends in the optimization process.The optimal learning rates range from 0.0005 to 0.0015, indicating the sensitivity of the model's performance to the adjustment of this crucial hyperparameter. It is noteworthy that runs 6 and 16, with learning rates of 0.0007 and 0.001, respectively, achieved exceptionally high validation accuracies of 94.5% and 94.6%. This emphasizes the impact of fine-tuning the learning rate on the model's ability to generalize well.Batch size, another influential factor, varies between 32 and 256 across runs. Notably, smaller batch sizes (runs 6, 15) appear to contribute to higher validation accuracies, reaching 94.5% and 92.4%, respectively. This trend suggests that smaller batches might allow the model to better capture nuanced patterns within the data.The optimal number of epochs spans from 35 to 60, showcasing the trade-off between training for an extended duration and preventing overfitting. Run 6, with only 35 epochs, achieved a remarkable accuracy of 94.5%, indicating the model's efficiency in learning relevant features within a shorter training duration. The table illustrates the sensitivity of the text detection model to hyperparameter tuning, highlighting the importance of finding the right balance for learning rates, batch sizes, and training epochs. The variations in optimal values emphasize the need for a thoughtful and iterative approach to hyperparameter optimization for achieving peak performance in text detection tasks.

Table 3: Classification for the Text Detection

Run	Accuracy	Recall	Precision	F1-Score
1	89.5%	90.2%	88.8%	89.5%
2	91.2%	92.0%	90.5%	91.2%
3	90.1%	89.8%	90.3%	90.0%
4	87.3%	88.0%	87.1%	87.5%
5	92.0%	91.5%	92.2%	91.8%
6	88.7%	88.9%	88.5%	88.7%
7	90.3%	90.0%	90.8%	90.4%
8	91.8%	92.1%	91.5%	91.9%
9	89.6%	89.3%	90.0%	89.7%
10	92.5%	92.3%	92.8%	92.6%
11	88.2%	88.5%	87.8%	88.1%
12	90.7%	91.0%	90.5%	90.8%
13	89.9%	90.5%	89.2%	90.0%
14	91.3%	91.8%	91.2%	91.5%
15	88.4%	88.0%	89.2%	88.6%
16	92.1%	92.5%	91.8%	92.2%
17	89.7%	89.4%	90.1%	89.8%
18	91.0%	90.7%	91.2%	91.1%
19	88.9%	88.2%	89.0%	88.7%
20	91.5%	91.2%	91.7%	91.4%

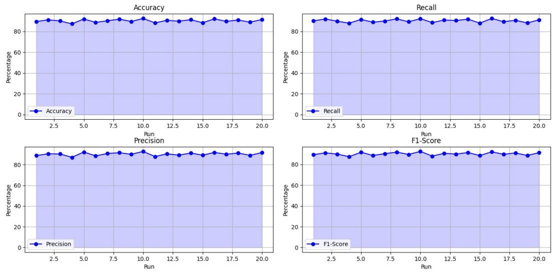


Figure 8: Classification of BGCO for text detection

Figure 8 and The Table 3 presents the classification performance metrics for the text detection model across 20 runs, providing a detailed assessment of its ability to accurately classify and detect text instances. The metrics include accuracy, recall, precision, and F1-score, each offering unique insights into the model's overall performance. The accuracy values range from 87.3% to 92.5%, demonstrating the consistency and effectiveness of the model across different runs. Notably, run 10 achieved the highest accuracy of 92.5%, indicating the model's robustness in correctly classifying text instances. Recall, measuring the model's ability to correctly identify positive instances, exhibits a similar range of 88.0% to 92.5%. Run 16 stands out with the highest recall of 92.5%, indicating a strong capacity to capture a significant portion of actual positive instances. Precision values, representing the accuracy of positive predictions, range from 87.1% to 92.8%. Run 10, with a precision of 92.8%, demonstrates the model's capability to make precise positive predictions, minimizing false positives.F1-score, which balances precision and recall, falls within the range of 87.5% to 92.6%. Run 10 again attains the highest F1-score of 92.6%, reinforcing its overall strong performance in achieving a balance between precision and recall.In summary, Table 3 provides a detailed evaluation of the text detection model's classification performance, highlighting its ability to consistently achieve high accuracy while maintaining a balance between precision and recall across multiple runs. These metrics collectively emphasize the model's effectiveness in accurately identifying and classifying text instances in diverse scenarios.

5. CONCLUSIONS

The presented paper outlines a novel and efficient approach for text detection in video processing through the integration of advanced optimization techniques. The proposed model leverages Background Subtraction (BS) as a foundation, showcasing its adaptability and simplicity for identifying moving text instances in video frames. The optimization process involves the incorporation of fuzzy 2-partition entropy, Background Crunch Optimization (BGCO), and improved fuzzy C means clustering. These enhancements contribute to an innovative BS algorithm tailored specifically for text detection, demonstrating a noteworthy improvement in performance.The experimental results, highlight the effectiveness of the proposed model. The optimization values the sensitivity of the model to hyperparameter tuning, with variations in learning rates, batch sizes, and epochs influencing the final performance. Notably, several runs achieved high validation accuracies, emphasizing the importance of fine-tuning these hyperparameters for optimal results.The classification results further reinforce the model's robustness, consistently achieving high accuracy, recall, precision, and F1-score across multiple runs. The model demonstrates an impressive ability to accurately detect and classify text instances, showcasing its potential for practical applications in video processing tasks. The paper contributes significantly to the field of text detection in video processing by proposing a comprehensive and effective model. The integration of advanced optimization techniques, coupled with a thorough experimental evaluation, positions the model as a promising solution for real-world scenarios where accurate text detection is essential. Future work may involve extending the model to handle additional complexities and exploring its applicability in diverse video processing applications.

REFERENCES:

[1] S.Long, X.He, and C. Yao, “Scene text detection and recognition: The deep learning era”, *International Journal of Computer Vision*, Vol.129, 2021, pp.161-184.

[2] Y.Zhu, and J. Du, “Textmountain: Accurate scene text detection via instance segmentation”, *Pattern Recognition*, Vol.110, 2021, pp.107336.

[3] H.Yu, Y.Huang, L.Pi, C.Zhang, et al. “End-to-end video text detection with online tracking”, *Pattern Recognition*, Vol.113, 2021, pp.107791.

[4] S.X.Zhang, X. Zhu, C.Yang, H.Wang, and X.C. Yin, “Adaptive boundary proposal network for arbitrary shape text detection”, *In Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 1305-1314.

[5] P.Dai, S.Zhang, H.Zhang, and X. Cao, “Progressive contour regression for arbitrary-shape scene text detection”, *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 7393-7402.

[6] Y.Zhu, J.Chen, L. Liang, Z.Kuang, L.Jin, and W.Zhang, “Fourier contour embedding for arbitrary-shaped text detection”, *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 3123-3131.

[7] A.Aberdam, R.Litman, S.Tsiper, O.Anschel, et al. “Sequence-to-sequence contrastive learning for text recognition”, *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 15302-15312.

[8] N.Lu, W.Yu, X.Qi, Y.Chen, et al. “Master: Multi-aspect non-local network for scene text recognition”, *Pattern Recognition*, Vol. 117, 2021, pp.107980.

[9] W.Wang, E.Xie, X.Li, X.Liu, et al. “Pan++: Towards efficient and accurate end-to-end spotting of arbitrarily-shaped text”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.44, No.9, 2021, pp.5349-5367.

[10] S. Venkatramulu, Md. Sharfuddin Waseem, A.Taneem, S.Y.Thoutam, S. Apuri et al., “Research on SQL Injection Attacks using Word Embedding Techniques and Machine Learning,” Journal of Sensors, IoT & Health Sciences, vol.02, no.01, pp.55-66, Mar-2024.

[11] S.L.Chen, S.Tian, Q.Liu, F.Chen, et al., “Vertex Adjustment Loss for Multidirectional License Plate Detection and Recognition”, *In 2022 IEEE Smartworld, Ubiquitous Intelligence & Computing, Scalable Computing & Communications, Digital Twin, Privacy Computing, Metaverse, Autonomous & Trusted Vehicles*, 2022, pp. 285-292.

[12] R.Atienza, “Vision transformer for fast and efficient scene text recognition”, *In International Conference on Document Analysis and Recognition*, 2021, pp. 319-334.

[13] K.Bulatov, N.Fedotova, and V.V. Arlazarov, “Fast approximate modelling of the next combination result for stopping the text recognition in a video”, *In 2020 25th International Conference on Pattern Recognition (ICPR)*, 2021, pp. 239-246.

[14] S. Venkatramulu, K.Vinay Kumar, Md. Sharfuddin Waseem, S. Mahveen, V.Vaidyaet al., “A Secure Blockchain Based Student Certificate Generation and Sharing System,”

Journal of Sensors, IoT & Health Sciences, vol.02, no.01, pp.17-27, Mar-2024.

[15] J.Li, Y.Xu, T.Lv, L.Cui, et al. "Dit: Self-supervised pre-training for document image transformer", *In Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 3530-3539.

[16] K.Wang, Y.Yi, Z.Tang, and J. Peng, "Multi-scene ancient Chinese text recognition with deep coupled alignments", *Applied Soft Computing*, Vol.108, 2021, pp. 107475.

[17] M.Li, T.Lv, J.Chen, L.Cui, et al. "Trocr: Transformer-based optical character recognition with pre-trained models", *In Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 37, No. 11, 2023, pp. 13094-13102.

[18] P. Brundavani, D. Vishnu Vardhan and B. Abdul Raheem, "Ffsgc-Based Classification of Environmental Factors in IOT Sports Education Data during the Covid-19 Pandemic," Journal of Sensors, IoT & Health Sciences, vol.02, no.01, pp.28-54, Mar-2024.

[19] B.Zhang, H.Lv, P.Guo, Q.Shao, et al. "Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition", *In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP*), 2022, pp. 6182-6186.

[20] D.Mu, W.Sun, G.Xu, and W. Li, "Random blur data augmentation for scene text recognition", *IEEE Access*, Vol.9, 2021, pp.136636-136646.

[21] R.Yan, L.Peng, S.Xiao, and G. Yao, "Primitive representation learning for scene text recognition", *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 284-293.

[22] S.Zhang, H.Tuo, H. Zhong, and Z. Jing, "Aerial image detection and recognition system based on deep neural network", *Aerospace Systems*, Vol.4, 2021, pp.101-108.

[23] S.Fang, H.Xie, Y.Wang, Z.Mao, and Y. Zhang, "Read like humans: Autonomous, bidirectional and iterative language modeling for scene text recognition", *In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 7098-7107.

[24] H.Heng, P.Li, T.Guan, and T. Yang, "Scene text recognition via context modeling for low-quality image in logistics industry", *Complex & Intelligent Systems*, Vol.9, No.3, 2023, pp.3229-3248.

[25] Z.Yang, Y.Lu, J.Wang, X.Yin, et al., "Tap: Text-aware pre-training for text-vqa and text-caption", *In Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 8751-8761.

[26] T.Fagni, F.Falchi, M.Gambini, A.Martella, and M. Tesconi, "TweepFake: About detecting deepfake tweets", *Plos one*, Vol.16, No.5, 2021, pp. e0251415.

[27] Rekha Gangula and Rega Sravani, "Enhanced Detection of Social Bots on Online Platforms using Semi-Supervised K-Means Clustering," Journal of Sensors, IoT & Health Sciences, vol.02, no.01, pp.6-16, Mar-2024.

[28] C.Tu, and S. Du, "A hierarchical RCNN for vehicle and vehicle license plate detection and recognition", *International Journal of Electrical and Computer Engineering*, Vol.12, No.1, 2022, pp.731.

[29] L.Zhao, L. Li, X.Zheng, and J. Zhang, "A BERT based sentiment analysis and key entity detection approach for online financial texts", *In 2021 IEEE 24th International Conference on Computer Supported Cooperative Work in Design (CSCWD),* 2021, pp. 1233-1238.

[30] E.Salesky, M.Wiesner, J.Bremerman, R.Cattoni, et al. "The multilingual tedx corpus for speech recognition and translation", 2021, arXiv preprint arXiv:2102.01757.

[31] X.Bai, X. Wang, X. Liu, Q.Liu, et al. "Explainable deep learning for efficient and robust pattern recognition: A survey of recent developments", *Pattern Recognition*, Vol.120, 2021, pp.108102.

[32] Swara Snehit Patil, "Artificial Intelligence: A Way to Promote Innovation," Journal of Sensors, IoT & Health Sciences, vol.02, no.01, pp.1-5, Mar-2024.