



Article

Assessing CNN Architectures for Estimating Correct Posture in Cruise Machinists

Fabian Arun Panaite ¹, Monica Leba ^{1,*}  and Andreea Cristina Ionica ² 

¹ Department of System Control and Computer Engineering, University of Petrosani, Universitatii 20, 332006 Petrosani, Romania; fabianpanaite@upet.ro

² Department of Management and Industrial Engineering, University of Petrosani, Universitatii 20, 332006 Petrosani, Romania; andreeaionica@upet.ro

* Correspondence: monicaleba@upet.ro or monicaleba@yahoo.com

Abstract: Cruise machinists operate in dynamic and physically demanding environments where improper posture can lead to musculoskeletal disorders, adversely affecting their health and work efficiency. Current ergonomic assessments in such settings are often generic and not tailored to the unique challenges of maritime operations. This paper presents a novel application of artificial intelligence tools for real-time posture estimation specifically designed for cruise machinists. The primary aim is to enhance occupational health and safety by providing precise, real-time feedback on ergonomic practices. We developed a dataset by capturing video recordings of cruise machinists at work, which were processed to extract skeletal outlines using advanced computer vision techniques. This dataset was used to train deep neural networks, optimizing them for accuracy in diverse and constrained shipboard environments. The networks were tested across various computational platforms to ensure robustness and adaptability. The AI model demonstrated high efficacy in recognizing both correct and incorrect postures under real-world conditions aboard ships. The system significantly outperformed traditional ergonomic assessment tools in terms of speed, accuracy, and the ability to provide instant feedback. The findings suggest that AI-enhanced ergonomic assessments could be a transformative approach for occupational health across various industries.

Keywords: posture estimation; SqueezeNet; GoogleNet; CPU; GPU; occupational safety



Citation: Panaite, F.A.; Leba, M.; Ionica, A.C. Assessing CNN Architectures for Estimating Correct Posture in Cruise Machinists. *Eng* **2024**, *5*, 1785–1803. <https://doi.org/10.3390/eng5030094>

Academic Editors: Antonio Gil Bravo and Dragan Pamucar

Received: 10 May 2024

Revised: 1 August 2024

Accepted: 1 August 2024

Published: 5 August 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Cruise ships are a popular mode of transportation and leisure, attracting millions of passengers each year. However, the cruise industry also employs a significant number of crew members, including machinists responsible for the operation and maintenance of the ship's engines and mechanical systems. Unfortunately, the work environment on cruise ships can be hazardous, and machinists are at a high risk of work-related accidents and injuries.

According to [1], the rate of work-related injuries among cruise ship machinists is significantly higher than the national average for the maritime industry. The study found that the most common types of injuries were musculoskeletal disorders, such as back injuries and sprains, as well as burns and lacerations. These injuries can have a significant impact on the health and well-being of the affected workers, as well as the overall efficiency and productivity of the cruise ship's operations.

To address this issue, cruise lines have implemented various prevention methods, such as providing personal protective equipment (PPE), implementing safety protocols, and offering training programs for machinists [2]. However, these approaches have had limited success in reducing the overall rate of work-related accidents. One potential solution to this problem is the use of AI-based training methods, which could provide more personalized and adaptive learning experiences for machinists, leading to improved safety outcomes.

Ref. [3] explored the use of AI-based training for maritime workers, including machinists. The researchers developed a virtual reality-based training system that used machine learning algorithms to assess the trainee's performance and provide personalized feedback and guidance. The results of the study showed that the AI-based training system was more effective in improving safety knowledge and skills compared to traditional training methods.

Given the significant risks faced by cruise ship machinists and the limitations of existing prevention methods, there is a clear need to explore more effective training solutions that leverage the power of AI. By developing and implementing AI-based training programs, cruise lines can potentially reduce the rate of work-related accidents, improve the health and safety of their employees, and enhance the overall efficiency and productivity of their operations.

This paper advocates for the adoption of an innovative solution: an AI Motion Analysis application. This application harnesses the power of machine learning algorithms and video capture technology to autonomously analyze and differentiate between optimal and suboptimal movements. By empowering trainees to independently practice and evaluate their techniques using readily available camera devices, this system eliminates the dependency on additional resources and human oversight.

Numerous studies have underscored the efficacy of training muscle memory, particularly when coupled with stability optimization strategies. Through the proposed application, training becomes not only more efficient but also accessible, as it can be conducted at the convenience of the trainee. Moreover, the objective nature of computer-based assessments ensures fair evaluations devoid of human subjectivity, thereby enhancing the overall effectiveness of training initiatives and safeguarding personnel health and safety.

The paper advances the application of convolutional neural networks (CNNs) for ergonomic posture evaluation, specifically tailored to the unique environment of cruise machinists. Our research directly targets the maritime industry, focusing on cruise machinists—a group often overlooked in ergonomic studies despite their high risk of posture-related injuries. The solutions we develop are not only theoretically robust but also practically applicable in improving workplace safety and ergonomics aboard cruise ships. By improving posture monitoring through advanced AI tools, our work supports the development of safer working environments and better health outcomes for a critical workforce.

To address the identified training gaps for cruise machinists, we compiled and processed a dataset using specialized software to extract frames from videos, augment images, identify key skeletal inflection points, and isolate skeletal outlines for training a neural network, ensuring unbiased learning across diverse human dimensions and proportions. The dual approach using two distinct CNN architectures—SqueezeNet and GoogleNet—allows us to explore and compare the efficacy of different models in a real-world setting, significantly contributing to the body of knowledge on practical CNN applications. Cross-platform training and evaluation not only tests the models' performance across different hardware but also makes our findings highly relevant for real-world deployments, where such flexibility in resource utilization is crucial.

Our study not only pushes forward the state of the art in applying deep learning to ergonomic assessments but also provides a tailored, robust, and versatile solution to a critical, industry-specific problem. We believe these contributions are of substantial importance to both the academic community and the maritime industry.

The subsequent sections of this paper continue as follows. Section 2: Literature Review offers an exploration of scientific literature about lifting techniques; muscle memory; factors influencing skill acquisition; and advancements in data acquisition, machine learning, and neural networks. Section 3: Materials and Methods is where the methodology employed in the study is presented, containing experimental procedures and data collection methods deployed to evaluate lifting posture correctness. Section 4: Experimental Design describes the careful attention devoted to crafting an experimental design characterized by effectiveness and efficiency. This phase involved the development of a robust system comprising

optimized hardware and software components tailored for seamless data collection and analysis. Section 5: Results and Discussion is where the findings of the study take center stage, accompanied by a thorough comparative analysis of the employed methods' metrics. Section 6: Conclusion is where the key findings and implications are synthesized, emphasizing the significance of the proposed AI Motion Analysis application in elevating workplace ergonomics and safety standards.

2. Related Work

Identifying correct posture in the workplace is essential for preventing musculoskeletal disorders (MSDs) and promoting overall well-being among workers. This literature review synthesizes recent research efforts aimed at evaluating posture and its implications for ergonomics and health.

The authors of [4] proposed a novel, vision-based, real-time method for evaluating postural risk factors associated with MSDs. By leveraging advanced computer vision techniques, their approach offers promising avenues for accurately assessing posture-related risks in real-world work settings. Similarly, an incremental Deep Neural Networks-based posture recognition model has been developed that is specifically tailored for ergonomic risk assessment in construction contexts [5]. The model demonstrates the potential of machine learning in providing precise and efficient solutions for identifying posture-related hazards.

Another work [6] explored ergonomic posture recognition using 3D view-invariant features from a single ordinary camera. The method capitalizes on innovative feature extraction techniques to overcome challenges related to viewpoint variations, paving the way for robust posture assessment systems. Complementing these technological advancements, insights from biomechanics studies are crucial for understanding the mechanical stability of the spine and its implications for injury and chronic low back pain [7,8].

Effective ergonomic interventions necessitate a comprehensive understanding of how various factors influence posture. For instance, lifting tasks pose significant risks to musculoskeletal health, with factors such as load knowledge and lifting height playing pivotal roles [9,10], for which there are proposed general lifting equations based on mechanical work during manual lifting, providing a theoretical framework for assessing lifting-related risks [11].

Furthermore, participatory ergonomic approaches have shown promise in reducing musculoskeletal exposure among workers [12]. By involving workers in the design and implementation of ergonomic interventions, these approaches ensure interventions are tailored to specific workplace contexts, thereby enhancing their effectiveness and acceptance.

In the construction industry, where non-routine work tasks are prevalent, reliable exposure assessment strategies are essential for identifying physical ergonomic stressors [13]. In [14], the authors conducted a postural analysis of building construction workers using ergonomic principles, highlighting the importance of addressing workplace-specific challenges to promote worker well-being.

Moreover, advancements in musculoskeletal modeling enable accurate estimation of spinal loading and muscle forces during lifting tasks, facilitating the design of ergonomic interventions tailored to individual needs [8]. Similarly, stretcher and backboard-related paramedic lifting tasks are ranked based on their biomechanical demands on the low back, providing insights for optimizing task design and training protocols [15].

Recent studies have also emphasized the importance of ergonomic analysis tools in identifying work-related MSDs, such as the efficacy of ergonomic analysis tools in industrial settings, highlighting their role in early detection and prevention of MSDs [16], and lifting hazards in a cabinet manufacturing company and proposed controls to mitigate risks associated with manual handling tasks [17].

Additionally, the accuracy of identifying low or high-risk lifting during standardized lifting situations was assessed, contributing to the development of reliable ergonomic assessment methods [18]. Acute effects of plyometric exercise on maximum squat performance in male athletes was examined, shedding light on the potential benefits of exercise interventions for improving posture and musculoskeletal health [19].

The literature reviewed underscores the multifaceted nature of identifying correct posture in the workplace. By integrating insights from computer vision, biomechanics, participatory ergonomics, and musculoskeletal modeling, researchers and practitioners can develop comprehensive strategies for mitigating posture-related risks and promoting occupational health and safety. Continued interdisciplinary research efforts are warranted to address evolving workplace challenges and ensure the well-being of workers across various industries.

However, these studies also exhibit certain limitations that underscore the novelty and importance of the proposed research on using AI tools for posture estimation in cruise machinists. Many of the related works focus on specific industries such as construction or healthcare; for example, ergonomic risk assessment for construction workers or the biomechanical demands on paramedics. This indicates a gap in research tailored to the unique environment of cruise ships, where space constraints, ship motion, and specific operational tasks might present unique challenges not addressed by these industry-specific studies. While papers such as [4,6] have incorporated advanced computer vision and machine learning techniques for posture recognition, these applications are generally in controlled or semi-controlled environments. The dynamic and often unpredictable environment aboard a cruise ship requires a robust AI system capable of handling diverse scenarios and lighting conditions, which may not be fully addressed by the existing models. The real-time vision-based method mentioned in paper [4] introduces the potential for immediate posture correction feedback, but it is not clear how well these systems perform in non-static, high-movement scenarios typical for cruise machinists. The research on AI tools for cruise machinists aims to fill this gap by developing models that not only assess posture in real-time but are also optimized for the highly mobile maritime context. Existing methods, such as those using 3D view-invariant features, attempt to address viewpoint variations. However, on a cruise ship, the varied and confined spaces could present extreme variations in viewpoint, which demands even more sophisticated adaptation from the AI models. Our research aims to develop an AI system that effectively manages these extreme variations, ensuring accurate posture assessment from any angle. While existing studies lay a strong foundation for using technology in ergonomic assessments, the proposed research on AI tools for posture estimation among cruise machinists addresses several unmet needs. It offers novel contributions by tailoring AI capabilities to the distinctive and challenging environment of cruise ships, ensuring that posture evaluation is accurate, comprehensive, and adaptable to highly variable operational contexts.

3. Materials and Methods

Despite the growing integration of AI tools across various sectors, as resulting from the literature review, there remains a notable research gap in their application specifically for the training of cruise machinists. While AI has been leveraged for operational efficiency and safety in many industrial settings, its potential to enhance the training programs for cruise machinists has not been fully explored. This gap highlights an opportunity to develop and implement AI-driven solutions that could significantly improve the training processes, safety standards, and ergonomic practices for machinists on cruise ships.

The problem to be addressed is the absence of AI-enhanced training solutions for cruise machinists, leading to suboptimal training outcomes and compromised safety and efficiency in cruise ship operations. The objective of this research is to develop and implement AI-based training tools designed for cruise machinists, aimed at enhancing the effectiveness of their training, improving safety compliance, and increasing operational efficiency on cruise ships. In order to achieve this objective, the methodology part contains the development steps of an AI-based tool. The paper ends with a comparative analysis in order to give the best suited solution for the identified problem, as shown in Figure 1.

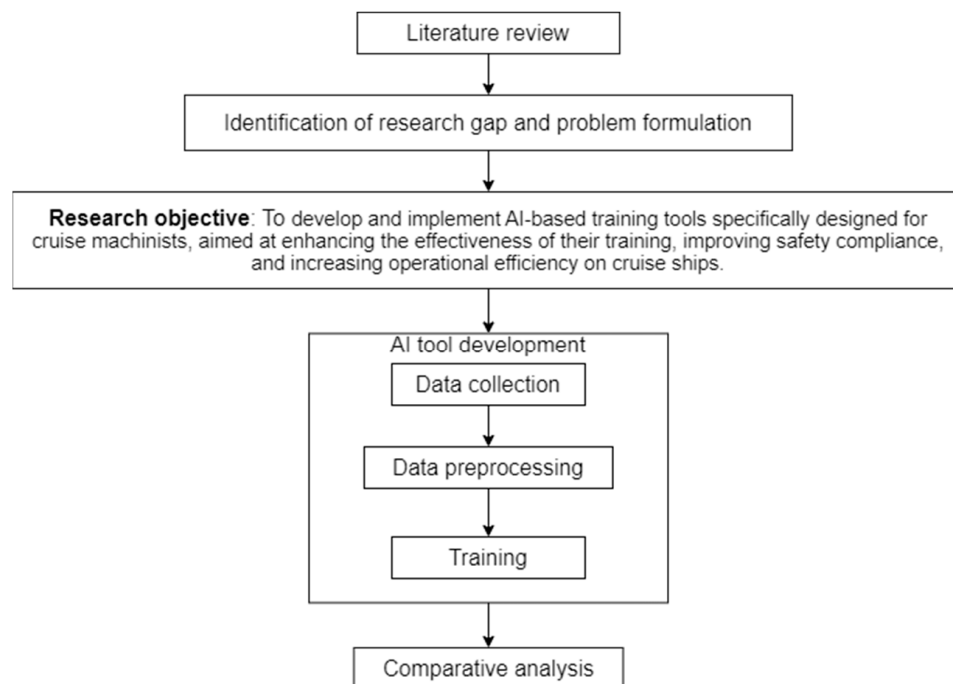


Figure 1. Methodology flowchart.

In this study, we employed pre-trained convolutional neural networks (CNNs) to recognize correct postures and positions during work for cruise machinists. Pre-trained CNNs are advantageous in this context due to their proven effectiveness in image recognition tasks, which allows for a significant reduction in the need for large amounts of labeled training data and diminishes training time. Leveraging these pre-existing networks that have been trained on vast and diverse datasets, such as ImageNet, enables the extraction of high-level features from the posture data captured by onboard cameras. These features are crucial for accurately categorizing postures as correct or incorrect based on ergonomic standards.

For this application, the selection of an appropriate pre-trained network is key, as it must efficiently handle the specific nuances of posture recognition in varied and dynamic maritime environments. The networks were fine-tuned on a smaller, domain-specific dataset composed of annotated images depicting various machinist lifting activities. This dataset included a range of postures captured under different conditions of lighting and background, mimicking the real-world scenario aboard cruise ships. The fine-tuning process involved adjusting the final layers of the networks so they become more specialized to the task-specific features of our ergonomic assessment criteria.

SqueezeNet offers an excellent model size and computational efficiency due to its use of squeeze and expand layers that reduce parameter count without a significant drop in accuracy. This makes SqueezeNet an attractive choice for deployment in environments where computational resources are limited, such as onboard systems on cruise ships. Despite its efficiency, SqueezeNet might struggle with lower accuracy in capturing highly detailed features compared to more complex models. Its performance can falter with very fine distinctions required in the ergonomic assessment of cruise machinists' postures.

GoogleNet, or Inception v1, introduces an inception module that uses different kernel sizes in the same layer, allowing it to capture information at various scales effectively. This ability makes GoogleNet particularly suitable for posture recognition, as it can detect details across diverse scenarios and varied scales, which are common in the unpredictable environments of cruise ships. The complexity of GoogleNet's architecture, while beneficial for feature detection, also means it requires more computational power and memory, potentially making it less ideal for deployment in resource-constrained onboard environ-

ments. Additionally, its complex structure may lead to longer training times during the fine-tuning phase.

Choosing the appropriate CNN architecture is critical in this project, balancing between computational efficiency and the ability to accurately recognize and analyze ergonomic postures. Each network offers distinct advantages and disadvantages, necessitating a careful consideration of the specific requirements and constraints of the setting. The analysis of these networks provides insights into optimizing model selection and tuning strategies to enhance posture recognition accuracy and operational efficiency aboard cruise ships.

In Figure 2 is presented the research flowchart that is further discussed.

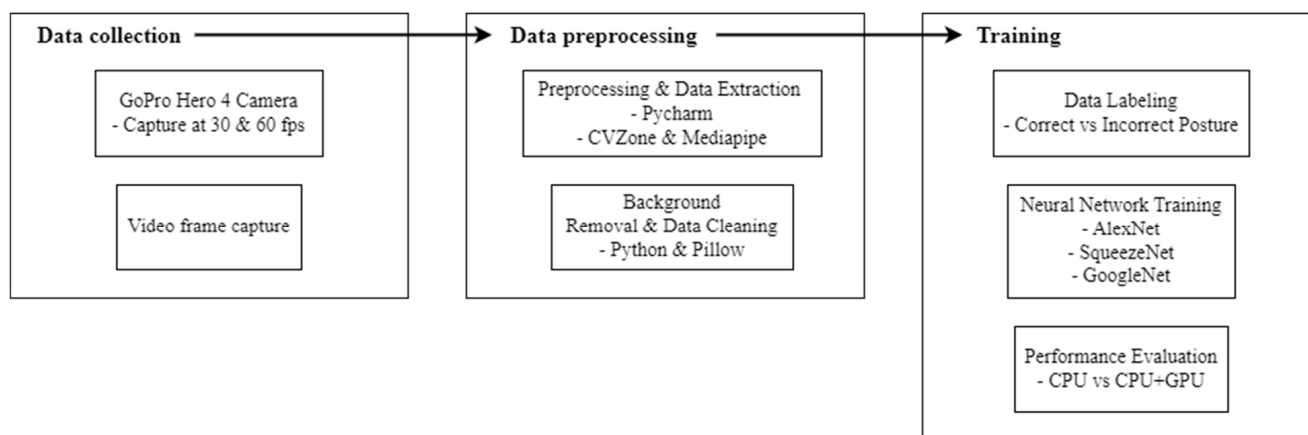


Figure 2. Research flowchart.

Data Collection: The research started with data acquisition using a GoPro Hero 4 camera, which captured the activities of cruise machinists at frame rates of 30 and 60 frames per second (fps). This device was chosen for its ability to record high-definition video in 720p MP4 format, allowing for clear and precise capture of detailed movements essential for the analysis. The versatility of this camera supported our objectives to observe and document complex machinist operations under varying light and motion conditions. In designing the system, it was crucial to integrate both effective and efficient technological solutions. The software stack for this project was centered around Pycharm, which facilitated the development environment for scripting and testing. Essential libraries such as CVZone and Mediapipe were integrated within the Pycharm environment to handle the capture and initial processing of motion data from the video files. Libraries from MATLAB were also employed, guaranteeing compatibility across different operating systems, thus maintaining the system's technological edge without being hindered by hardware resource discrepancies.

Data Preprocessing: Following data collection, video frames were extracted and subsequently preprocessed (Figure 3). During this phase, each frame was processed through a Python (version 3.10.12) script which utilized non-intrusive techniques to extract 3D view-invariant relative joint positions and angles. This method ensured that the data remained independent of camera viewpoint variations, which is critical for accurate posture analysis in a dynamic environment. The application prototype used in this project incorporated the Python libraries within a script to refine the capture of worker motion. The motion data extracted then underwent a thorough cleaning process using another Python script leveraging the Pillow library. This script removed backgrounds from captured images, replacing them with a simplified 8-bit color (dark blue in our case) to enhance the clarity of the skeleton representation of the body positions. After the cleaning and extraction phase, each frame was analyzed to remove any with invalid data or missing elements. At this stage, all data were labeled into two categories: correct posture and incorrect posture. This labeled data then formed the basis for training the neural networks.

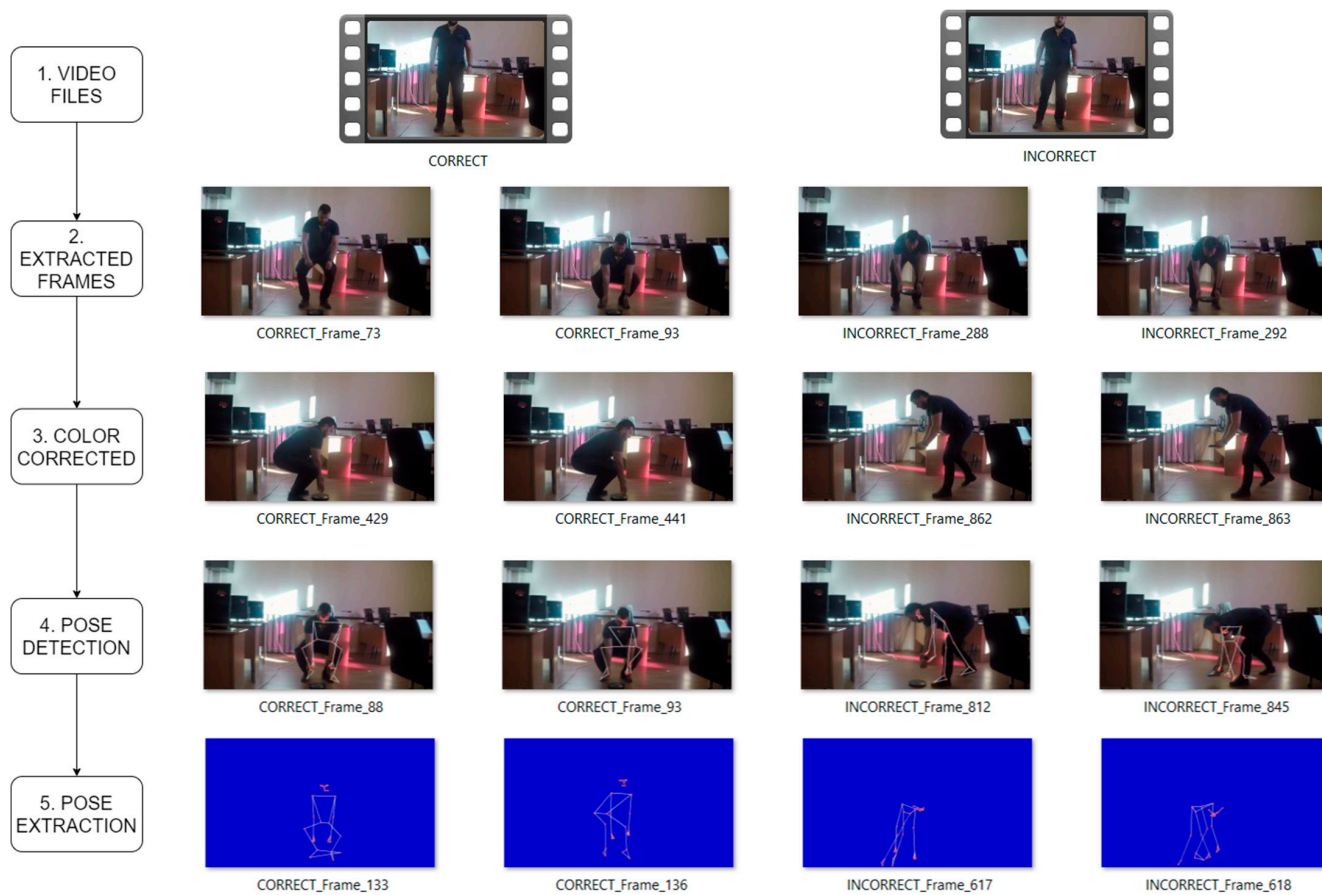


Figure 3. Dataset preprocessing flow.

Dataset Analysis: The dataset critical for our study comprised video footage specifically selected to capture a comprehensive array of postural dynamics of cruise machinists engaged in various lifting tasks. These activities were chosen for their relevance in assessing the ergonomic risks associated with routine maritime mechanical operations. To ensure the robustness and accuracy of the training data, a meticulous preprocessing protocol was followed. Initially, the raw video footage was processed to isolate key frames that accurately represent typical postures machinists assume during their work. This selection process was guided by both the frequency of specific postures in the footage and their significance from an ergonomic perspective. Subsequent to frame selection, a detailed annotation process was undertaken. Each frame was labeled as “CORRECT” or “INCORRECT” based on stringent ergonomic criteria developed by occupational health experts. These criteria were established based on current best practices in ergonomic assessments, ensuring that the labels reflect meaningful distinctions in posture quality that could impact machinist health and safety. The annotation process involved multiple expert reviewers to minimize subjectivity and enhance the reliability of the posture classifications. This dual-review system ensured that each frame’s label was as objective as possible, providing a solid foundation for the subsequent machine learning training process. The consistency and accuracy of these labels are crucial, as they directly influence the learning outcomes of the neural network models. The labeled dataset not only serves as the basis for training our CNN models but also plays a critical role in validating their effectiveness in real-world ergonomic assessments.

Training: We employed the two pre-trained networks—SqueezeNet and GoogleNet—for their suitability to the task. SqueezeNet and GoogleNet demonstrated over 90% accuracy, confirming their effectiveness for the project’s needs. Both networks were further trained

using configurations that employed just the CPU, as well as combinations of the CPU and GPU, to compare performance metrics under different computational loads.

SqueezeNet has a total of 18 layers, structured to maximize efficiency and performance. The architecture begins with a standard convolutional layer with 96 filters of size 7×7 , followed by a ReLU activation and a max-pooling layer. This is followed by eight Fire modules, which consist of a squeeze layer and an expand layer. The Fire modules are arranged in the following sequence: Fire module 2, Fire module 3, Fire module 4 (followed by a max-pooling layer), Fire module 5, Fire module 6, Fire module 7 (followed by a max-pooling layer), Fire module 8, and Fire module 9. After the Fire modules, there is another convolutional layer with 1000 filters of size 1×1 , followed by a ReLU activation. The architecture concludes with a global average pooling layer, which averages the spatial dimensions of the feature maps, and a softmax classifier that outputs the probability distribution over 1000 classes.

GoogleNet, also known as Inception v1, has a complex architecture with 22 layers, designed to achieve high performance in image classification tasks. The architecture begins with two standard convolutional layers followed by max-pooling layers. The core innovation of GoogleNet is the Inception module, which allows the network to capture multi-scale information by performing 1×1 , 3×3 , and 5×5 convolutions in parallel within each module, followed by a max-pooling layer. These outputs are concatenated along the channel dimension. The network includes nine Inception modules stacked linearly, interspersed with max-pooling layers to reduce the spatial dimensions. Following the Inception modules, the architecture ends with an average pooling layer, a dropout layer, a fully connected layer with 1000 units, and a softmax classifier for outputting the probability distribution over the classes. Additionally, GoogleNet includes auxiliary classifiers connected to intermediate layers to improve gradient flow and provide regularization during training. This design significantly reduces the number of parameters while maintaining high accuracy.

To modify SqueezeNet and GoogleNet to output two classes instead of their default number of classes, the final layers of each network were adjusted to accommodate binary classification. For SqueezeNet, modifying the last convolutional layer in the classifier section was necessary to change the output to two classes, ensuring that the network outputs two class scores by reducing the dimensionality of the feature maps to match the number of classes. Similarly, for GoogleNet (Inception v1), the adjustment involved editing the number of outputs in the final fully connected layer to output only two scores instead of the original 1000. This modification required unlocking and editing the final layer's configuration to reduce the number of class scores. Both networks were then fine-tuned on the new binary classification dataset to optimize performance for the new task.

This methodological approach not only ensured the high accuracy and relevance of the posture recognition system but also underscored the necessity for continuous technological adaptation and enhancement to meet and exceed the demanding standards of modern work ergonomics. By integrating advanced computational methods and machine learning techniques, this research illustrates the potential of AI-driven solutions to significantly improve occupational safety in dynamic environments such as those encountered by cruise machinists.

4. Experiment

4.1. Experiment Design

The central aim of this study was to assess and compare the efficacy of two advanced CNN models, SqueezeNet and GoogleNet, when trained across two different computational platforms: CPUs and GPUs. The objective focused on determining the optimal model and hardware combination for accurately estimating correct postures in cruise machinists, who are particularly susceptible to ergonomic risks due to the physically demanding nature of their work.

The dataset integral to this research was derived from video recordings that captured a broad spectrum of postural dynamics of cruise machinists engaged in lifting tasks.

Advanced image processing techniques were employed to preprocess this footage, enhancing frame quality and isolating the subjects from their environments. This involved segmentation to separate the subjects from the background, normalization to standardize the intensity of images, and augmentation to artificially expand the dataset under varied conditions, simulating real-world scenarios.

Each frame extracted during preprocessing was meticulously annotated as “CORRECT” or “INCORRECT”, based on ergonomic guidelines provided by occupational health experts. This annotation was crucial for training the models to recognize and differentiate between ergonomically safe and risky postures. A total of 1006 data points were annotated, with a distribution of 559 labeled as “CORRECT” and 447 as “INCORRECT” (Figure 4). This data formed the backbone of our analysis, serving as the basis for training and validating the CNN models.

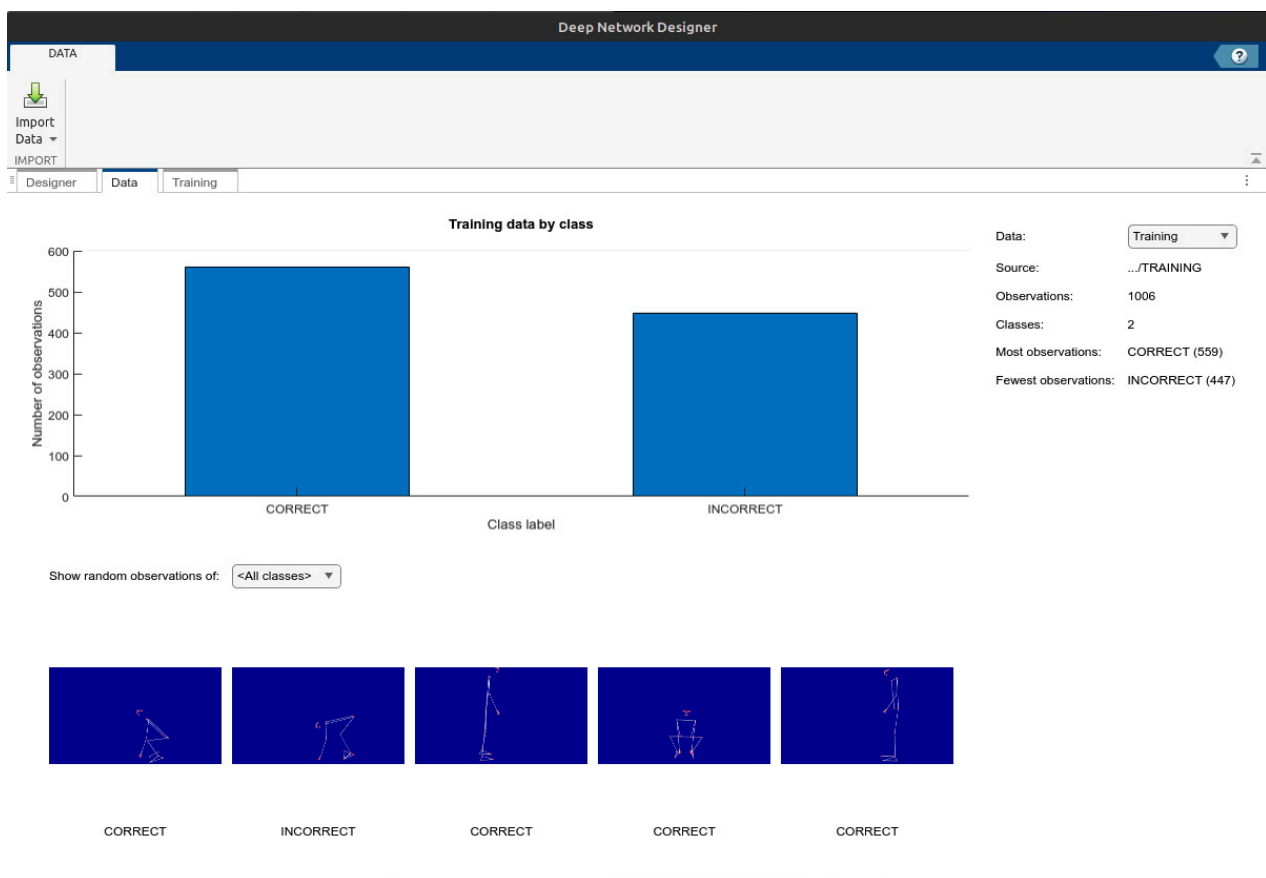


Figure 4. Dataset analysis.

Both SqueezeNet and GoogleNet were subjected to structured training and validation phases. Each model was trained on 80% of the data and validated on the remaining 20%. The training was conducted with batch sizes and learning rates adjusted according to the computational limits of the CPUs and GPUs to ensure consistent learning conditions. The models underwent training over multiple epochs, ceasing only when no significant improvement in validation accuracy was observed, to prevent overfitting and to ensure comprehensive learning.

The experiment utilized mid-range to high-end commercial CPUs and GPUs to reflect a realistic work scenario. This choice was intended to provide insights applicable to potential real-world applications and to test the models under conditions that might be encountered in an actual work setting.

The performance of each model was systematically evaluated using a variety of metrics including accuracy, precision, recall, and F1-score. These metrics were chosen to provide a

holistic view of each model's capability to accurately classify postures as "CORRECT" or "INCORRECT". Additionally, training efficiency was assessed by monitoring the duration and computational resource utilization during the model training sessions.

To facilitate a detailed analysis of the training outcomes, the Deep Network Designer software (version 23.2) was employed. This tool helped visually represent the structure and distribution of the dataset, as well as the training progress of each model, enabling a comprehensive evaluation of the models' performance across different computational platforms.

We employ the Cross-Entropy Loss as our loss function because our task involves classification. This function evaluates the model's performance by comparing the predicted labels with the actual labels. The general formula for Cross-Entropy Loss is:

$$J = -\frac{1}{N} \sum_{i=1}^N \sum_{c=1}^C y_{ic} \log(\hat{y}_{ic}) \quad (1)$$

where

N represents the total number of samples in the current batch;

C represents the total number of classes in the classification problem;

y_{ic} is a binary indicator that is 1 if the actual class of sample i is c , and 0 if it is not, and essentially marks the true label of each sample;

\hat{y}_{ic} is the predicted probability that the model assigns to the sample i being in class c . These probabilities are the output of a softmax function at the final layer of the network.

For our models, the formula for Cross-Entropy Loss is defined as:

$$J = -\frac{1}{10} \sum_{i=1}^{10} (y_{i1} \cdot \log(\hat{y}_{i1}) + y_{i2} \cdot \log(\hat{y}_{i2})) \quad (2)$$

We use as optimizer Stochastic Gradient Descent with Momentum (SGDM), that is an optimization algorithm used for training neural networks, namely a variant of gradient descent where the model parameters are updated using a small subset (mini-batch) of the training data, rather than the entire dataset. This approach helps in speeding up the training process and allows for more frequent updates of the model parameters. The momentum is an enhancement to the basic SGD algorithm that helps accelerate the convergence of the training process. It does this by adding a fraction of the previous update to the current update. This helps in smoothing out the oscillations in the gradients and leads to faster convergence.

In mathematical terms, the update rule for SGD with momentum uses the following formulas:

$$v_{t+1} = \mu v_t - \alpha \nabla_{\theta} J(\theta_t) \quad (3)$$

$$\theta_{t+1} = \theta_t + v_{t+1} \quad (4)$$

where

v_t is the accumulated past gradients, scaled by the momentum coefficient, and helps to smooth out the gradient updates and accelerates convergence;

μ is the momentum coefficient that determines the contribution of the past gradients, for which the value in our models is 0.9;

α is the learning rate, which controls the size of the steps taken towards the minimum of the loss function, for which the value in our models is 0.001;

$\nabla_{\theta} J(\theta_t)$ is the gradient of the loss function with respect to the parameters θ at the current time step t and indicates the direction and rate of change of the loss function;

θ_t represents the current values of the model parameters;

θ_{t+1} represents the updated values of the model parameters after applying the velocity term.

This experimental design was crafted to rigorously assess the implications of each model's performance across varied hardware setups. By aligning the experimental approach closely with practical demands, this study aims to significantly advance the appli-

cation of AI in enhancing occupational health and safety standards, providing valuable insights for the deployment of AI-driven posture estimation tools in work environments.

4.2. Experiment Implementation

For SqueezeNet, the training dashboard visualizations were instrumental in monitoring the model’s progress and efficacy throughout the training phase on a GPU, as depicted in Figure 5. The accuracy trends, initially starting at lower levels, showed a consistent increase across iterations, eventually stabilizing above 90%. This upward trajectory in both training and validation accuracies, highlighted by circular indicators at significant points, emphasized the model’s capacity to generalize well beyond the training data, achieving a final validation accuracy of 91.32%.

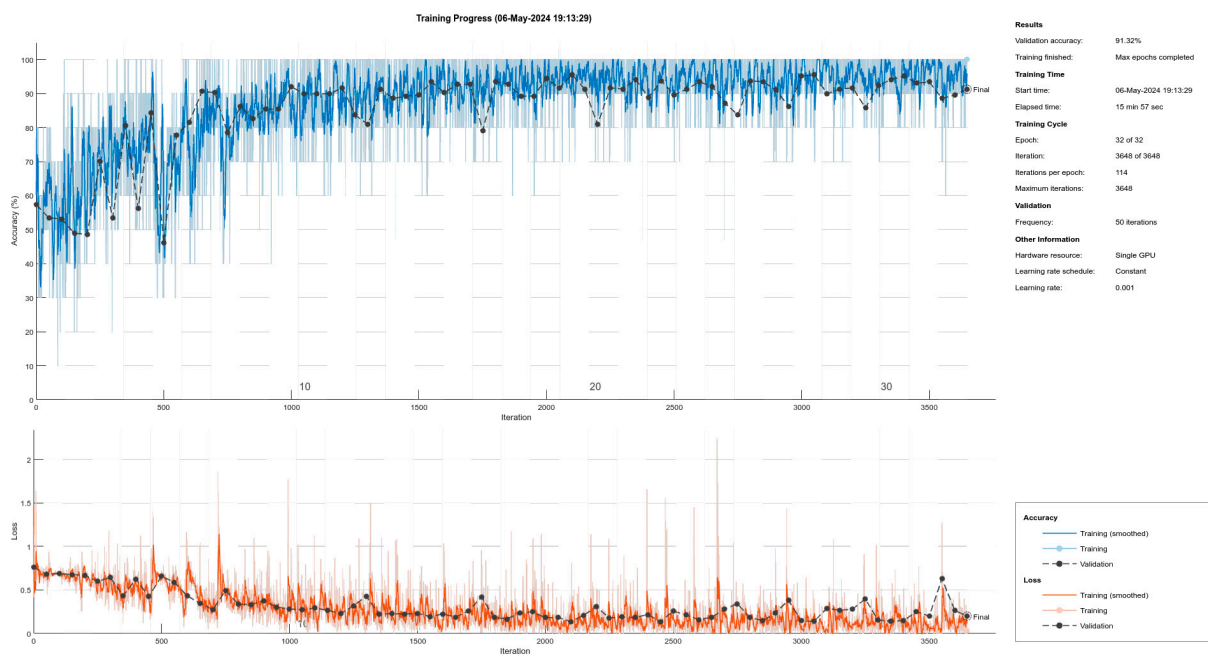


Figure 5. Training SqueezeNet on GPU.

Parallel to this, the loss trends presented in the lower graph of the dashboard and in Table 1 delineated a continuous decrease in both training and validation loss, indicating effective learning and model convergence. The model underwent a rigorous training regimen, consisting of 32 epochs with 114 iterations each, culminating in a total of 3648 iterations. This training was executed efficiently on a single GPU, taking approximately 15 min and 57 s, underscoring the GPU’s capacity to facilitate rapid computational processing.

Table 1. Final losses.

	SqueezeNet on CPU	SqueezeNet on GPU	GoogleNet on CPU	GoogleNet on CPU
Final training loss	0.6	0.5	0.25	0.3
Final validation loss	0.4	0.4	0.2	0.25

In contrast, the training dashboard for SqueezeNet on a CPU, shown in Figure 6, documented a notably higher final validation accuracy of 95.14%. This improvement suggests that the extended training duration, which totaled 44 min and 51 s, allowed for more thorough optimization of model weights, despite the inherently slower processing capabilities of CPUs.

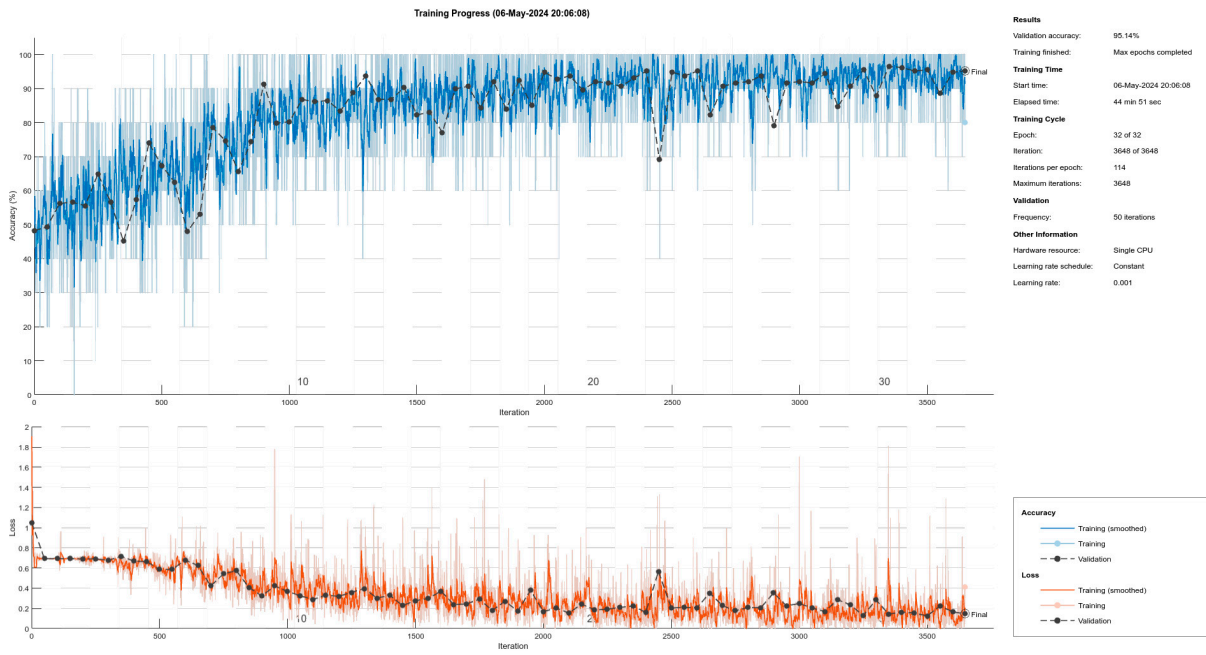


Figure 6. Training SqueezeNet on CPU.

The loss metrics for the CPU model (Table 1) similarly demonstrated a steady decline, mirroring the GPU’s performance but with the added benefit of achieving a higher validation accuracy. This points to the CPU’s effectiveness in handling deep learning tasks where precision is important, albeit at a slower pace.

The training of GoogleNet provided further insights into the performance disparities between CPU and GPU environments. As shown in Figure 7, the GPU-trained GoogleNet model displayed a quick improvement in validation accuracy, peaking at 97.22% within just 19 min and 26 s. The loss trends followed a declining trajectory, consistent with an effective learning process (Table 1).

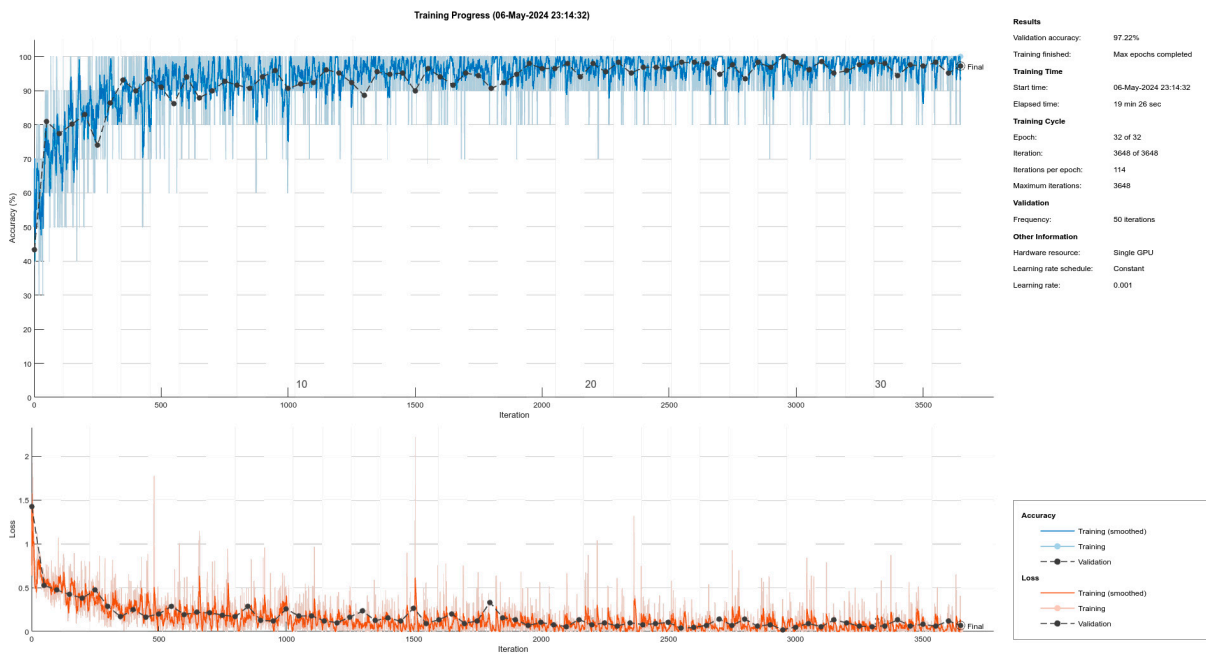


Figure 7. Training GoogleNet on GPU.

Conversely, as illustrated in Figure 8, the GoogleNet model trained on a CPU achieved a slightly higher validation accuracy of 97.57%, albeit over a considerably longer duration of 89 min and 29 s. This extended period allowed for deeper model refinement, which was particularly evident in the stabilization of accuracy and loss rates at the later stages of training (Table 1).

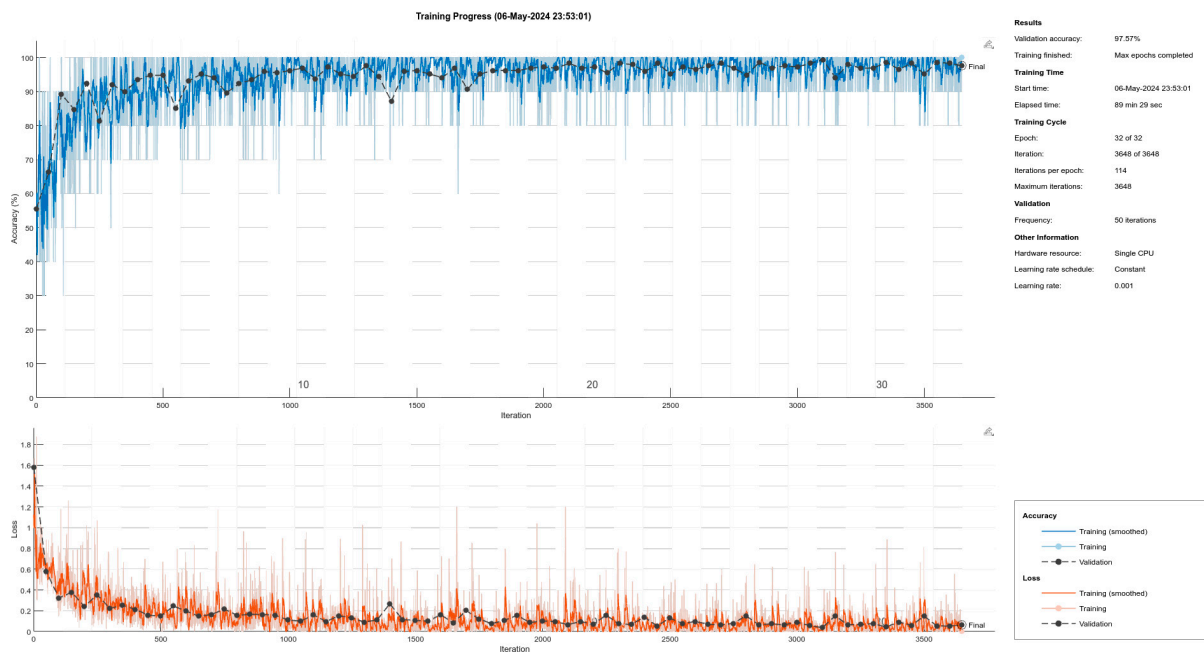


Figure 8. Training GoogleNet on CPU.

The experiment underscored the critical trade-offs between training on CPUs and GPUs. While GPUs offered speed and efficiency, CPU training resulted in marginally higher accuracies, demonstrating the importance of selecting hardware based on specific performance needs and time efficiencies. This comparative analysis between the computational platforms provides essential insights into the deployment strategies for deep learning models, emphasizing the need to balance speed, scalability, and precision based on available resources and project requirements.

This experiment not only highlighted the strengths and limitations of each model across different hardware but also reinforced the significance of strategic hardware selection in optimizing deep learning implementations for practical, real-world applications.

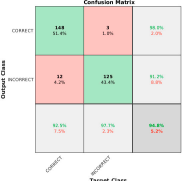
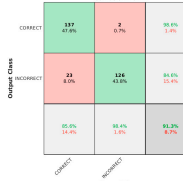
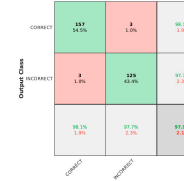
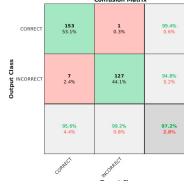
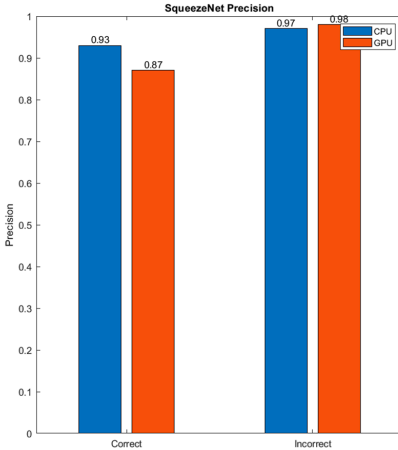
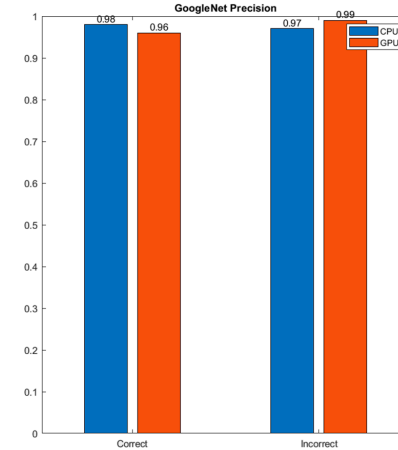
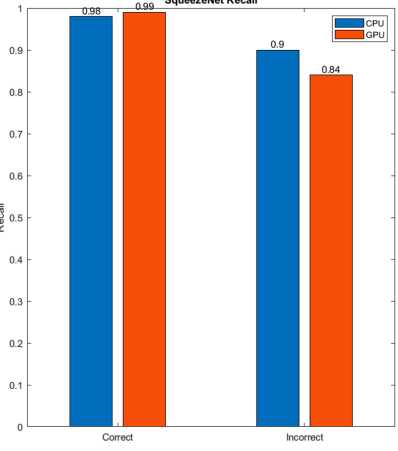
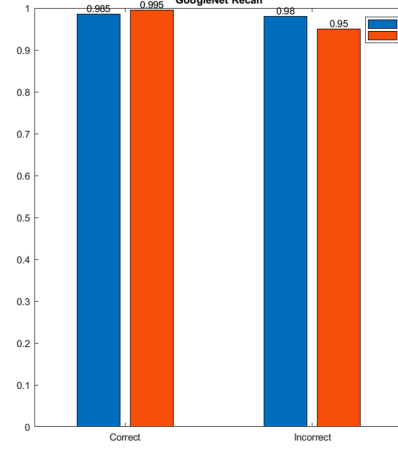
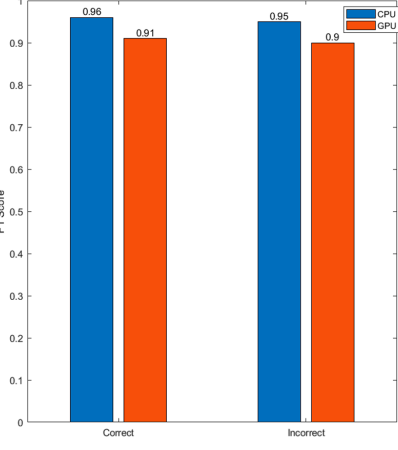
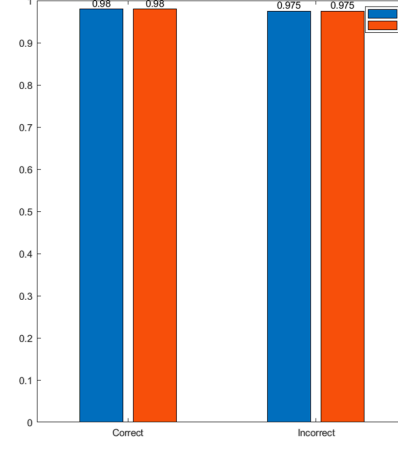
5. Results and Findings

In the evaluation phase of our study, the performance of the two types of pre-trained CNNs, SqueezeNet and GoogleNet, was assessed. These networks were trained on both GPU and CPU hardware settings to determine their efficacy and efficiency under different computational conditions. An overview of the comparison of the performance of the two CNNs based on the resulting metrics is presented in Table 2, followed by detailed comparative analysis.

5.1. Comparative Analysis of SqueezeNet Training on CPU and GPU

This evaluation specifically focuses on various metrics including the confusion matrix, precision, recall, and F1-score for each class (Table 2). This analysis is critical, as it helps identify the optimal operational conditions for the SqueezeNet model, ensuring that it is not only efficient in terms of computational resource usage but also maintains high accuracy in practical applications. Through this investigation, we seek to understand the trade-offs involved in deploying deep learning models across different hardware platforms, providing insights into how each platform supports the model's ability to classify accurately and reliably.

Table 2. Performance comparative metrics.

	SqueezeNet on CPU	SqueezeNet on GPU	GoogleNet on CPU	GoogleNet on GPU																																																
Confusion matrix	 <table border="1"> <caption>Confusion Matrix</caption> <tr> <td>Correct</td> <td>138 (93.8%)</td> <td>1 (0.7%)</td> <td>9 (6.2%)</td> </tr> <tr> <td>Incorrect</td> <td>12 (8.2%)</td> <td>129 (87.6%)</td> <td>9 (6.2%)</td> </tr> <tr> <td></td> <td>95.8%</td> <td>95.7%</td> <td>96.8%</td> </tr> </table>	Correct	138 (93.8%)	1 (0.7%)	9 (6.2%)	Incorrect	12 (8.2%)	129 (87.6%)	9 (6.2%)		95.8%	95.7%	96.8%	 <table border="1"> <caption>Confusion Matrix</caption> <tr> <td>Correct</td> <td>137 (93.6%)</td> <td>2 (0.7%)</td> <td>9 (6.4%)</td> </tr> <tr> <td>Incorrect</td> <td>23 (15.8%)</td> <td>126 (86.2%)</td> <td>9 (6.4%)</td> </tr> <tr> <td></td> <td>95.6%</td> <td>95.4%</td> <td>91.3%</td> </tr> </table>	Correct	137 (93.6%)	2 (0.7%)	9 (6.4%)	Incorrect	23 (15.8%)	126 (86.2%)	9 (6.4%)		95.6%	95.4%	91.3%	 <table border="1"> <caption>Confusion Matrix</caption> <tr> <td>Correct</td> <td>137 (93.5%)</td> <td>3 (0.8%)</td> <td>9 (6.3%)</td> </tr> <tr> <td>Incorrect</td> <td>9 (6.2%)</td> <td>128 (87.5%)</td> <td>9 (6.3%)</td> </tr> <tr> <td></td> <td>95.1%</td> <td>97.2%</td> <td>97.9%</td> </tr> </table>	Correct	137 (93.5%)	3 (0.8%)	9 (6.3%)	Incorrect	9 (6.2%)	128 (87.5%)	9 (6.3%)		95.1%	97.2%	97.9%	 <table border="1"> <caption>Confusion Matrix</caption> <tr> <td>Correct</td> <td>133 (91.3%)</td> <td>1 (0.7%)</td> <td>9 (6.4%)</td> </tr> <tr> <td>Incorrect</td> <td>7 (4.8%)</td> <td>127 (87.3%)</td> <td>9 (6.4%)</td> </tr> <tr> <td></td> <td>95.6%</td> <td>95.2%</td> <td>97.2%</td> </tr> </table>	Correct	133 (91.3%)	1 (0.7%)	9 (6.4%)	Incorrect	7 (4.8%)	127 (87.3%)	9 (6.4%)		95.6%	95.2%	97.2%
Correct	138 (93.8%)	1 (0.7%)	9 (6.2%)																																																	
Incorrect	12 (8.2%)	129 (87.6%)	9 (6.2%)																																																	
	95.8%	95.7%	96.8%																																																	
Correct	137 (93.6%)	2 (0.7%)	9 (6.4%)																																																	
Incorrect	23 (15.8%)	126 (86.2%)	9 (6.4%)																																																	
	95.6%	95.4%	91.3%																																																	
Correct	137 (93.5%)	3 (0.8%)	9 (6.3%)																																																	
Incorrect	9 (6.2%)	128 (87.5%)	9 (6.3%)																																																	
	95.1%	97.2%	97.9%																																																	
Correct	133 (91.3%)	1 (0.7%)	9 (6.4%)																																																	
Incorrect	7 (4.8%)	127 (87.3%)	9 (6.4%)																																																	
	95.6%	95.2%	97.2%																																																	
Precision	 <p>Model Precision Comparison on CPU vs GPU</p> <table border="1"> <caption>SqueezeNet Precision</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.93</td> <td>0.87</td> </tr> <tr> <td>Incorrect</td> <td>0.97</td> <td>0.98</td> </tr> </table>	Category	CPU	GPU	Correct	0.93	0.87	Incorrect	0.97	0.98	 <p>Model Precision Comparison on CPU vs GPU</p> <table border="1"> <caption>GoogleNet Precision</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.98</td> <td>0.96</td> </tr> <tr> <td>Incorrect</td> <td>0.97</td> <td>0.99</td> </tr> </table>	Category	CPU	GPU	Correct	0.98	0.96	Incorrect	0.97	0.99																																
Category	CPU	GPU																																																		
Correct	0.93	0.87																																																		
Incorrect	0.97	0.98																																																		
Category	CPU	GPU																																																		
Correct	0.98	0.96																																																		
Incorrect	0.97	0.99																																																		
Recall	 <p>Model Recall Comparison on CPU vs GPU</p> <table border="1"> <caption>SqueezeNet Recall</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.98</td> <td>0.99</td> </tr> <tr> <td>Incorrect</td> <td>0.9</td> <td>0.84</td> </tr> </table>	Category	CPU	GPU	Correct	0.98	0.99	Incorrect	0.9	0.84	 <p>Model Recall Comparison on CPU vs GPU</p> <table border="1"> <caption>GoogleNet Recall</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.995</td> <td>0.995</td> </tr> <tr> <td>Incorrect</td> <td>0.98</td> <td>0.95</td> </tr> </table>	Category	CPU	GPU	Correct	0.995	0.995	Incorrect	0.98	0.95																																
Category	CPU	GPU																																																		
Correct	0.98	0.99																																																		
Incorrect	0.9	0.84																																																		
Category	CPU	GPU																																																		
Correct	0.995	0.995																																																		
Incorrect	0.98	0.95																																																		
F1-Score	 <p>Model F1 Score Comparison on CPU vs GPU</p> <table border="1"> <caption>SqueezeNet F1 Score</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.96</td> <td>0.91</td> </tr> <tr> <td>Incorrect</td> <td>0.95</td> <td>0.9</td> </tr> </table>	Category	CPU	GPU	Correct	0.96	0.91	Incorrect	0.95	0.9	 <p>Model F1 Score Comparison on CPU vs GPU</p> <table border="1"> <caption>GoogleNet F1 Score</caption> <tr> <th>Category</th> <th>CPU</th> <th>GPU</th> </tr> <tr> <td>Correct</td> <td>0.98</td> <td>0.98</td> </tr> <tr> <td>Incorrect</td> <td>0.975</td> <td>0.975</td> </tr> </table>	Category	CPU	GPU	Correct	0.98	0.98	Incorrect	0.975	0.975																																
Category	CPU	GPU																																																		
Correct	0.96	0.91																																																		
Incorrect	0.95	0.9																																																		
Category	CPU	GPU																																																		
Correct	0.98	0.98																																																		
Incorrect	0.975	0.975																																																		
Training time	44 min 51 s	15 min 57 s	89 min 29 s	19 min 26 s																																																
Training accuracy	95.14%	91.32%	97.57%	97.22%																																																

The confusion matrices offer a detailed view into the performance of the SqueezeNet CNN trained on two different hardware setups: CPU and GPU. These matrices are critical for understanding not just the overall accuracy but also the model's ability to correctly classify observations into either the "CORRECT" or "INCORRECT" class.

The comparison between the two hardware platforms indicates some crucial differences. The CPU seems to provide a balanced performance, achieving high accuracy in classifying both "CORRECT" and "INCORRECT" instances. This balance might be due to the CPU's ability to manage the training process more steadily over time, allowing for more stable learning dynamics. On the other hand, the GPU, despite its faster processing capabilities and ability to handle more extensive data manipulations efficiently, shows a tendency to yield slightly lower precision in classifying "INCORRECT" cases.

Both training environments show high overall accuracy, but the choice between CPU and GPU may depend on specific requirements of the classification task. For applications where minimizing false positives is crucial, CPU-based training might be preferable due to its slightly better balance in handling both classes. In scenarios where speed and handling large datasets are more critical, and slight variations in false-positive rates are acceptable, GPU-based training would be advantageous. This analysis demonstrates the importance of selecting the appropriate hardware based on the specific performance metrics crucial to the task's success.

The evaluation of the SqueezeNet model trained on both CPU and GPU platforms reveals consistency in performance across various metrics—specifically, precision, recall, and F1-score. This analysis compares these key performance indicators to assess how each training environment affects the model's ability to accurately classify observations into "CORRECT" and "INCORRECT" classes.

In terms of precision, both models show slightly higher scores for the "INCORRECT" class compared to the "CORRECT" class. This suggests that both the CPU and GPU models are marginally better at correctly identifying negative instances without mistakenly classifying positive instances as negative.

Regarding recall, both the CPU and GPU models demonstrate higher recall scores for the "CORRECT" class compared to the "INCORRECT" class. This indicates a tendency of both models to capture a majority of the actual "CORRECT" instances, albeit at the risk of missing some "INCORRECT" instances.

Both the CPU and GPU-trained models exhibit high F1 scores for each class, nearing 1.0. This indicates a well-balanced performance, where both models effectively handle the classification tasks with minimal disparity between the classes. The similar F1 scores across both platforms suggest that regardless of the computational power, the model maintains a robust balance between precision and recall, ensuring effective overall performance. The consistency in performance metrics across CPU and GPU training suggests that SqueezeNet is relatively stable and not overly sensitive to the underlying hardware when it comes to classification accuracy. Both platforms manage to sustain a balance between identifying true positives and avoiding false positives, albeit with slight variations in their precision and recall rates across different classes.

Given the specific requirements of the application, which involve identifying and flagging incorrect lifting positions of heavy objects, and considering the evaluated metrics, SqueezeNet trained on a CPU demonstrates superior results. This finding underscores the model's robustness in accurately detecting and classifying incorrect postures, an essential feature for enhancing workplace safety and reducing ergonomic risks among workers engaged in physically demanding tasks.

5.2. Comparative Analysis of GoogleNet Training on CPU and GPU

This comparative analysis is essential for understanding how each platform influences the model's accuracy and its ability to classify instances into "CORRECT" and "INCORRECT" categories efficiently. Through this examination, we seek to highlight the strengths

and limitations of each training setup, providing insights into the optimal deployment scenarios for the GoogleNet model based on specific performance criteria.

The confusion matrices for GoogleNet trained on CPU and GPU provide insightful data into the performance characteristics of each hardware environment. Analyzing these matrices helps to understand how each platform handles the classification of “CORRECT” and “INCORRECT” classes under similar training conditions but different computational resources (Table 2).

The confusion matrix from the model trained on a CPU shows a higher overall accuracy compared to the GPU, with a very high correct classification rate for both “CORRECT” and “INCORRECT” classes. These results indicate a strong performance with minimal error rates, suggesting that CPU training, despite generally being slower, can achieve high accuracy in classification tasks.

Conversely, the GPU-trained model, while also performing well, exhibited a slight decrease in accuracy. The overall pattern suggests that while the GPU model is very effective in correctly identifying “INCORRECT” instances, it is slightly less reliable for “CORRECT” classifications compared to the CPU model.

In both cases, the models show very low false-positive rates for the ‘INCORRECT’ class, which is crucial in applications where it is important not to mistakenly label an instance as “INCORRECT”. However, the GPU model seems to be more prone to false negatives in the “CORRECT” class than the CPU model.

Both the CPU and GPU-trained models exhibit very high F1 scores close to 1.0 for both classes. This indicates an excellent balance of precision and recall, suggesting that the models are well-tuned and perform robustly in distinguishing between “CORRECT” and “INCORRECT” classifications. The consistency of high F1 scores across both hardware platforms highlights the model’s capacity to maintain performance stability regardless of the computational environment.

In terms of precision, both models demonstrate high accuracy, with scores nearing perfection. Notably, both models exhibit slightly higher precision for the “INCORRECT” class compared to the “CORRECT” class. This trend suggests that both the CPU and GPU models are particularly effective at avoiding false positives when identifying “INCORRECT” instances, which is crucial for applications where the cost of incorrectly rejecting a correct instance is high.

As for recall, both models again perform very well, with the “CORRECT” class showing marginally higher recall than the “INCORRECT” class.

The parallel performance of GoogleNet across the CPU and GPU environments in terms of precision and recall underscores a consistent model behavior, which is an excellent indicator of the model’s robust generalization capabilities. Both models manage to achieve high precision and recall, ensuring that the trade-offs between these metrics are minimized. This consistency is especially noteworthy given the different processing powers and architectures of CPUs and GPUs, which can often lead to variations in training dynamics and outcomes.

Considering the specific needs of the application along with the performance metrics obtained, GoogleNet trained on a GPU emerges as the more suitable option, due to the shorter training time. This approach leverages the GPU’s capability to handle the model’s complexity effectively, ensuring optimal performance in accurately identifying critical postural assessments, which is crucial for the application’s success.

Moreover, the dataset comprises 1006 images, of which 559 are labeled as “CORRECT” and 447 as “INCORRECT”. This represents a slight imbalance, with approximately a 10% difference between the two classes. Observations during training indicate that this imbalance does not adversely affect the F1 score of the models, and the difference in accuracy between the two classes remains below 5% across most configurations. An exception was noted with the SqueezeNet model trained on a GPU, where the accuracy for the “INCORRECT” class was nearly 15% higher than that for the “CORRECT” class. Despite the minor discrepancy in the number of images between the classes, this imbalance

does not negatively impact the precision of the models. This finding underscores the robustness of the training process and the models' ability to handle slight variations in class distribution effectively.

6. Conclusions

Lifting heavy objects is the most widespread activity among cruise ship crews, and the integration of AI tools into the training protocol is a significant advancement in promoting safe and efficient work practices.

The current training protocol for cruise machinists includes both demonstration and practical exercises. Simply put, a qualified instructor demonstrates the correct technique for lifting heavy objects, such as a toolbox. The instructor stands with feet a shoulder-width apart, positions themselves close to the object, and bends their knees slightly while keeping the back straight and head up. They then lift the object using leg strength, keeping it close to the body. Following this demonstration, each machinist practices lifting a similar object, receiving real-time feedback and corrections to ensure proper form and technique. This hands-on approach enables machinists to apply the principles demonstrated, fostering the development of muscle memory essential for safe lifting practices. Supervision is provided by a certified safety instructor or an experienced crew member with specialized training in ergonomics and heavy lifting techniques.

The newly developed AI tool enhances this training protocol by providing additional, complex support. It can observe and analyze the posture of multiple machinists simultaneously across different workstations, offering instant feedback and corrective advice. This capability allows the AI tool to effectively augment the role of safety instructors, enabling more comprehensive and efficient training sessions. By integrating this AI technology, the training process not only becomes more scalable but also gains a higher level of precision in monitoring and improving the ergonomic practices of crew members.

When comparing the performance of SqueezeNet trained on a CPU and GoogleNet trained on a GPU for the task of identifying incorrect lifting postures by cruise machinists, several distinctions emerge that are crucial for application-specific decisions. Both models exhibit high levels of accuracy and F1-scores, demonstrating their robustness in classification tasks. However, GoogleNet on a GPU shows a particularly strong performance, making it the preferable choice for activities involving the lifting of heavy objects.

While SqueezeNet offers consistent performance across both CPU and GPU platforms and excels in environments with limited computational resources due to its efficient architecture, it does not match the slightly superior metrics achieved by GoogleNet. Specifically, GoogleNet's recall for the "CORRECT" class is notably higher, suggesting its enhanced capability to capture all relevant instances without missing many. This feature is vital in settings where missing a correct instance could result in increased risk of injury, making high recall an essential metric.

In terms of training efficiency, GoogleNet, despite its complex architecture requiring longer training times on CPUs, benefits significantly from the accelerated processing capabilities of GPUs. This harnessing of GPU capabilities allows GoogleNet to maintain high precision and accuracy, crucial for the demanding requirements of ergonomics in occupational health. On the other hand, the quicker training times of SqueezeNet, while advantageous in constrained environments, may not always meet the stringent accuracy requirements needed for precise ergonomic assessment.

The higher validation accuracy and precision of GoogleNet when trained on a GPU confirm its suitability for critical applications where the stakes of incorrect classification are high, such as in ensuring the safety of cruise machinists during heavy lifting tasks. Thus, for projects where accuracy and the ability to handle complex scenarios are paramount, and where computational resources allow, GoogleNet on a GPU stands out as the better option, outweighing the benefits of SqueezeNet's speed and efficiency in resource-constrained settings. This analysis highlights the importance of aligning the model selection and

training platform with specific operational goals and constraints, ensuring the deployment of the most effective AI solution for the task at hand.

The study presented significant insights into the use of AI for posture evaluation, yet it also revealed certain limitations that could direct future research efforts. One of the primary limitations involves the image preprocessing technique which renders the skeletal outline against a blue background. This blue background, while visually distinct, occupies a significant portion of the image. The extensive presence of this uniform background could potentially slow down the process by including many irrelevant elements when discriminating between “CORRECT” and “INCORRECT” postures. Such a background might inadvertently influence the AI’s learning and operational efficiency, as the system spends computational resources processing areas of the image that do not contribute to posture analysis.

Additionally, the potential for generating false “INCORRECT” results in scenarios where a subject is standing with a slight forward lean due to an anatomically poor posture poses a challenge. This misclassification can lead to incorrect training or feedback, potentially reinforcing bad habits rather than correcting them. This limitation underscores the need for more refined posture recognition capabilities that can differentiate between intentional poses and those resulting from an individual’s physical limitations.

Regarding the dataset, its enhancement could significantly improve the model’s accuracy. Currently, the dataset could benefit from including a broader range of images capturing a variety of subjects interacting with different types of objects varying in shape, size, and weight. Expanding the dataset in this manner would not only provide a richer set of training data but also enhance the model’s ability to generalize across real-world scenarios, which is critical for the robust performance of AI systems in diverse operational settings.

Looking forward to further research, there are several paths to explore. One potential improvement could involve the enhancement of the image preprocessing step. This could include thickening the lines that form the subject’s outline in the images, which may aid in more accurate and efficient feature extraction by the AI. Additionally, incorporating an algorithm for a bounding box that isolates only the part of the image containing the outline could streamline the processing. This modification would ensure that the AI focuses solely on the relevant features of posture, thereby improving both the speed and accuracy of the posture evaluation process. Such advancements would address the current limitations and deploy more effective and efficient AI tools for ergonomic assessment in various industrial settings.

Author Contributions: Conceptualization, M.L. and A.C.I.; methodology, A.C.I.; software, F.A.P.; validation, M.L., F.A.P. and A.C.I.; formal analysis, M.L.; investigation, A.C.I.; resources, F.A.P.; data curation, F.A.P.; writing—original draft preparation, F.A.P.; writing—review and editing, M.L. and A.C.I.; visualization, A.C.I.; supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Ethical review and approval were waived for this study due to the use of fully anonymized images where individual participants cannot be identified.

Informed Consent Statement: Informed consent was obtained from all subjects involved in the study.

Data Availability Statement: Data available upon request at https://github.com/Ph-Ai-bian/UPET_V_DATASET.

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

References

1. Abdallah, S.; Genaidy, A.; Karwowski, W.; Shell, R.; Sonbol, A.; Ravelo, E. Theoretical basis for general lifting equations based on mechanical work performed during manual lifting. *Theor. Issues Ergon. Sci.* **2007**, *8*, 551–564. [[CrossRef](#)]
2. Armstrong, D.P.; Makhoul, P.J.; Sinden, K.E.; Fischer, S.L. Ranking stretcher and backboard related paramedic lifting tasks based on their biomechanical demand on the low back. *IIEE Trans. Occup. Ergon. Hum. Factors* **2019**, *7*, 9–19. [[CrossRef](#)]
3. Brandt, M.; Madeleine, P.; Samani, A.; Jakobsen, M.D.; Skals, S.; Vinstrup, J. Accuracy of identification of low or high risk lifting during standardised lifting situations. *Ergonomics* **2017**, *61*, 710–719. [[CrossRef](#)] [[PubMed](#)]
4. Capodaglio, E.M. Participatory ergonomics for the reduction of musculoskeletal exposure of maintenance workers. *J. Occup. Environ. Hyg.* **2020**, *17*, 376–386. [[CrossRef](#)] [[PubMed](#)]
5. Cholewicki, J.; McGill, S.M. Mechanical stability of the in vivo lumbar spine: Implications for injury and chronic low back pain. *Clin. Biomech.* **1996**, *11*, 1–15. [[CrossRef](#)] [[PubMed](#)]
6. Jensen, R.L.; Ebben, W.P. Acute effects of plyometric exercise on maximum squat performance in male athletes. *J. Strength Cond. Res.* **2003**, *17*, 68–71.
7. Johnson, L.; Williams, K. Evaluating the effectiveness of safety protocols for cruise ship machinists. *Safety Sci.* **2019**, *117*, 123–132.
8. Kim, H.-K.; Zhang, Y. Estimation of lumbar spinal loading and trunk muscle forces during asymmetric lifting tasks: Application of whole-body musculoskeletal modelling in OpenSim. *Ergonomics* **2016**, *59*, 563–576. [[CrossRef](#)] [[PubMed](#)]
9. Kulkarni, V.S.; Devalkar, R.V. Postural analysis of building construction workers using ergonomics. *Int. J. Constr. Manag.* **2018**, *18*, 464–471. [[CrossRef](#)]
10. Lee, S.; Park, J. AI-based training for maritime workers: A virtual reality approach. In Proceedings of the International Conference on Intelligent Transportation Systems (ITSC), Macau, China, 8–12 October 2021; pp. 1234–1239.
11. Li, L.; Martin, T.; Xu, X. A novel vision-based real-time method for evaluating postural risk factors associated with musculoskeletal disorders. *Appl. Ergon.* **2020**, *87*, 103138. [[CrossRef](#)] [[PubMed](#)]
12. Liu, J.; Qu, X.; Liu, Y. Influence of load knowledge on lifting biomechanics. *J. Occup. Environ. Hyg.* **2022**, *19*, 230–235. [[CrossRef](#)] [[PubMed](#)]
13. Ngo, B.P.T.; Yazdani, A.; Carlan, N.; Wells, R. Lifting height as the dominant risk factor for low-back pain and loading during manual materials handling: A scoping review. *IIEE Trans. Occup. Ergon. Hum. Factors* **2017**, *5*, 158–171. [[CrossRef](#)]
14. Paquet, V.; Punnett, L.; Woskie, S.; Buchholz, B. Reliable exposure assessment strategies for physical ergonomics stressors in construction and other non-routinized work. *Ergonomics* **2007**, *50*, 1200–1219. [[CrossRef](#)] [[PubMed](#)]
15. Pascual, S.A.; Naqvi, S. An investigation of ergonomics analysis tools used in industry in the identification of work-related musculoskeletal disorders. *J. Occup. Environ. Hyg.* **2015**, *5*, 237–245. [[CrossRef](#)] [[PubMed](#)]
16. Schneider, S.; Grant, K.A.; Habes, D.J.; Bertsche, P.K. Ergonomics: Lifting hazards at a cabinet manufacturing company: Evaluation and recommended controls. *Appl. Occup. Environ. Hyg.* **1997**, *12*, 253–258. [[CrossRef](#)]
17. Smith, J.; Jones, M. Work-related injuries among cruise ship machinists. *J. Occup. Environ. Med.* **2020**, *62*, 345–352.
18. Zhang, H.; Yan, X.; Li, H. Ergonomic posture recognition using 3D view-invariant features from a single ordinary camera. *Autom. Constr.* **2018**, *93*, 233–243. [[CrossRef](#)]
19. Zhao, J.; Obonyo, E. Applying incremental Deep Neural Networks-based posture recognition model for ergonomics risk assessment in construction. *Adv. Eng. Inform.* **2021**, *50*, 101374. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.