MuayTech: Deep Learning for Action Recognition in Muay Thai Strikes

Teddy Danielson[1], **Loni Halsted-Ruelas**[2], **Rintaro Oshima**[3], **Arushi Tyagi**[4], **and Anthony Wong**[5]

[1]**University of California, Santa Cruz**

ABSTRACT

This paper describes the development of a robust deep learning pipeline to classify Muay Thai strikes using video footage. Existing literature and models demonstrate a lack of strike label complexity and datasets specific to Muay Thai.[8] Starting with a dataset of 21 manually labeled clips, we developed a pipeline that scaled this dataset to 410 clips and further augmented it to 1,640 clips. Early experiments with skeleton-based action recognition using *MMAction2* [7] were limited by format constraints, leading us to adopt video-based transformer methods. Fine-tuning *VideoMAE* [11] and *TimeSformer* [2] models on *Kinetics-400* [5] allowed us to classify 14 Muay Thai strike types. Additionally, we addressed label confusions by flattening categories into 8 broader strike classes and using focal loss to handle class imbalances. Our final *TimeSformer* [2] model achieved a testing accuracy of 70.18%, significantly improving upon existing methods and setting a new benchmark for Muay Thai action recognition.

Keywords: Muay Thai, Action Recognition, Deep Learning, TimeSformer, VideoMAE, Computer Vision, Data Augmentation

INTRODUCTION

Problem

Sports classification using machine learning has been gaining significant attention as it provides players with valuable feedback to improve their skills and make better judgments during practice and competition. In combat sports like boxing and kickboxing, several machine learning models have been developed to classify actions such as kicks and punches. However, these models have notable limitations. Most of them are restricted to specific sports like boxing or general kickboxing, while Muay Thai, which features a broader range of techniques, remains largely underexplored. Existing models for Muay Thai often struggle to classify the extensive variety of strikes used in the sport.

One example is the *StrikeMetrics* [9] model, which attempted to classify punches and kicks in Muay Thai matches. However, it only achieved an accuracy of 12.5% on real Muay Thai fight data. This poor performance is likely due to the model's inability to distinguish between specific types of strikes in Muay Thai, such as different styles of punches, kicks, or knee strikes. These shortcomings highlight the need for a more robust and sport-specific classification model that can handle the unique complexities of Muay Thai.

Goal

To address the limitations of existing models, our project aims to develop a machine learning model capable of accurately classifying all distinct types of strikes in Muay Thai. Unlike previous models that treat all punches and kicks as general categories, this model will provide a more fine-grained classification of Muay Thai strikes. By distinguishing between different types of punches (e.g., jab, cross, hook) and kicks (e.g., roundhouse, teep), our model aims to offer fighters, coaches, and fighting leagues more detailed feedback that can be used for training, performance analysis, and skill development.

The development of this classification model is expected to significantly improve the practicality of AI-based tools for Muay Thai training. Fighters and coaches will be able to receive feedback on specific strike techniques, allowing for more targeted improvement. This will bridge the gap between current machine learning models and the real-world demands of Muay Thai training and judging.

Approach

Our approach to building this classification model involves leveraging advances in deep learning. We plan to utilize and fine-tune pre-trained pose estimation models, adapting them to recognize the unique movements of Muay Thai. The model will detect key points of the human body (like joints and limbs) from fight footage and use this information to classify 14 specific types of strikes.

Key technical components of our approach include:

- **Transfer Learning:** Using existing pose estimation models and fine-tuning them with Muay Thai-specific data.
- **Data Augmentation:** Expanding the dataset by applying techniques like rotation, flipping, and frame interpolation to improve model robustness.

By combining these techniques, we aim to develop a system that can generalize well to real-world Muay Thai fight videos, even in the presence of visual occlusions, fast movements, and varying video angles.

METHODOLOGY

Data Handling

Sourcing Data

This project sourced its data from existing datasets, footage of experts practitioners found online, and amateur recorded footage by experienced members. The datasets used were Human Motion Database (HMDB51) for its kick and punch classes and UCF101 for its Punch, Boxing-Punching-Bag and Boxing-Speed Bag action classes [6][10]. These datasets provided coarse classification and hundreds of clips to annotate into our desired classes. Footage was considered ”Expert Footage” if it featured retired or active professional Muay Thai fighters or coaches demonstrating technique. Most of the data was sourced from Sylvie von Douglass Ittu who has published footage of retired Muay Thai Fighters shadow boxing and demonstrating technique [12]. Amateur footage was recorded by group members performing strikes on a bag and person from a variety of camera angles.

Annotation

Data was annotated by hand, where time stamps of strikes were identified and then labeled based on the strike demonstrated. The tool used for video annotation was VGG Image Annotator (VIA) [3]. When deciding strike criteria, we considered a strike finished when it reached full extension and resembled the strikes shown in Figure 2. This criterion was chosen because oftentimes when strikes occurred in a row the next strike would begin as the preceding strike was being retracted. As such a strike ending at full extension prevented multiple strikes from being captured in a single time stamp. A strike was considered starting when the body began moving towards the final positions. For example a punch began when the strike side hip began rotating forward and the opposite hip moved back. For kicks, it began as the striking leg began to be raised off the floor and make its way towards its target.

Dataset Creation

The dataset began with 21 hand-labelled video clips. To scale this process, we developed a dataset creation pipeline after transitioning away from *MMAction2* [7]. This pipeline took VIA’s JSON annotation output and segmented full length videos into label specific clips. These clips were then organized into a structured UCF-formatted directory where each folder corresponded to a specific strike type (e.g., ”jab,” ”cross”). This pipeline enabled the rapid expansion of the dataset to 410 labeled clips. An additional function was created to create train/test/validation sets out of the dataset. UCF format was chosen to take advantage of existing UCF dataloaders and functions implemented in Pytorchvision.

Inference Pipeline

Our inference pipeline took another approach to video segmentation. In a real-world scenario, our model would be used to analyze and label a consistent sequence of strikes from a single video (i.e., one angle of one fight). This isn’t possible with the UCF-formatted directory, as clips are collected by type and not source. To test performance on a real-world example, we developed script that would take in a single video and timestamp information and spit out a collection of *unlabled* clips. Those clips were then directly fed into the model, which output labeled predictions. Those predictions were then manually audited for accuracy.

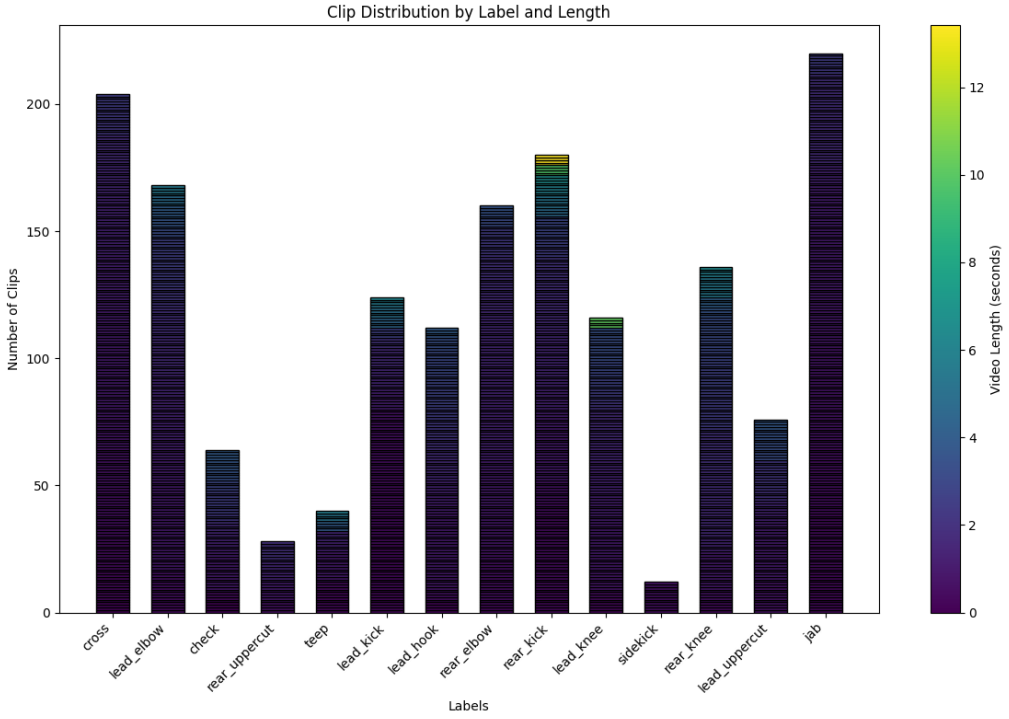


Figure 1. Clip Distribution by Label and Length

Data Augmentation

To address the initial size limitation of the dataset, we implemented a robust data augmentation pipeline. Techniques included:

- **Random Cropping:** Frames were cropped to introduce spatial variability.
- **Flipping:** Horizontal flips simulated mirrored movements.
- **Temporal Augmentation:** Frames were dropped or duplicated to vary the sequence.
- **Adding Grain:** Gaussian noise simulated real-world imperfections.

These augmentations increased the dataset size from 410 to 1,640 clips, improving model robustness.

Label Flattening

Confusion matrices revealed frequent misclassification of strikes with their opposite-side counterparts (See Figure 3). To mitigate this, we introduced a flattened label set of 8 categories, merging side-differentiated strikes (e.g., ”jab” and ”cross” became ”straight punches”). Figure 2 displays the flattened label set. The 14-strike label set include differentiation on sides on all positions except teep and check.

Model Selection, Development, and Hyperparamter Optimization

MMAction2

Our early experiments with *MMAction2* [7] relied on PKL files generated by *Mediapipe* [13] for skeleton-based pose estimation. However, the format constraints and limited applicability of MMAction2 for strike classification led to a shift toward video-based inputs.

VideoMAE

The shift in data annotation methods required a change in model architecture as well to accomodate the lack of Skeleton Keypoint data. The first model we began training and fine-tuning on our novel dataset was Video Masked Auto Encoder or *VideoMAE* [11]. We chose this model as a jumping off point because of its expansive documentation and Hugging Face implementation. The specific model architecture

(a) Straight Punch **(b)** Hook **(c)** Uppercut

(d) Kick **(e)** Elbow **(f)** Knee

(g) Teep **(h)** Check

Figure 2. Examples of the 8 strike categories in the flattened label set. Each image corresponds to a representative strike from the respective category [4].

we settled on was a VideoMae model fine tuned on the *Kinetics-400* [5] dataset which is an expansive action classification database with 400 distinct action classifications. This specific model has 81 Million parameters, a GeLu activation function and a fully connected classifier.

The trainer implementation took advantage of the pytorchvideo library for dataset importation and the pytorchvision library for basic video augmentations. The specific dataset importation was done using the UCF dataloader, which automatically loaded in UCF formatted data into train/test/validation dataloaders. Pytorchvideo was also used to implement the frame sampling approach in which we utilized uniform frame sampling and a frame sample rate of 4. This sampling approach meant that for each clip the model would get frames from all parts of the video. The basic augmentation was a randomized video flip and blurring applied to the data with a 0.5% probability of application.

Training made use of the hugginface trainer and trainingargument functions, with a standardized learning rate of 5e-5, an Adam W optimizer, Cross Entropy Loss with a batch of 32 and a varied dropout rate. The evaluation metric was Accuracy with additional confusion matrices computed in order analyze where our model misclassified strikes and development intuition into why these classification errors occurred. Batch size was determined based on GPU memory utilization which sat at around 85% on an A100 which was required as a result of the high computational cost and training time.

The model was trained on two versions of the dataset one with only 400 or so clips without augmentations applied, and because of its smaller size we trained for 20 epochs to try and develop intuition on model performance, learning rate, potential over fitting issues and as a baseline to see how long we should train future models. However, this proved computationally very expensive taking 2-3 hours, even with our high utilization on a A100 GPU. As such once we had created our augmented dataset of around 1500 clips we had to drastically reduce our epoch training scheme as training just 5 epochs took 3+ hours on the same hardware. The time consuming nature of training this model coupled with lackluster results made it hard to efficiently tune multiple hyperparameters. This coupled with poor results discussed later led to an eventual switch to our final model architecture.

TimeSformer

Despite its compatibility, *VideoMAE's* prolonged training times and difficulty in tuning hyperparameters prompted us to explore the *TimeSformer* [2] model. *TimeSformer* provided better computational efficiency and enhanced accuracy, making it a better fit for our task.

The *TimeSformer* model was similarly trained on our UCF-formatted directory of strikes, both with 410 originally-sourced clips, and 1640 post-augmentation clips. An A100 GPU was utilized for this training as well, taking significantly less time ($\approx$ 20 minutes). This allowed us to experiment in the hyperparameter space, using several different combinations until we determined optimal set for our purposes.

To avoid overfitting-an issue found early in *TimeSformer* exploration-a dropout rate of 30% was applied to encourage generalization. To address label class imbalance, we implemented focal loss: a loss function down-weighted heavily-represented labels and emphasized harder-to-classify ones, helping improve performance on underpresented strikes and positions. An Adam optimizer with a learning rate of 1×10^{-4}. Weight decay was implicitly handled by the optimizer. To further help avoid overfitting, a scheduler, *ReduceLROnPlateau*, was used to dynamically adjust the learning rate based on validation loss trends. If no improvement was observed for two epochs, learning rate would be reduced by a factor of .1. In addition to this, early stopping was implemented to aid the anti-overfitting effort. Training was terminated if validation loss was not improved by *min_delta*= 0.001 after two consecutive epochs. 10 epochs were used to start but early stoppage would occur at 6.

Our final model version was tested on a flattened label set, original label set, augmented dataset, and original dataset, allowing us to compare the effect that those processes had on our results.

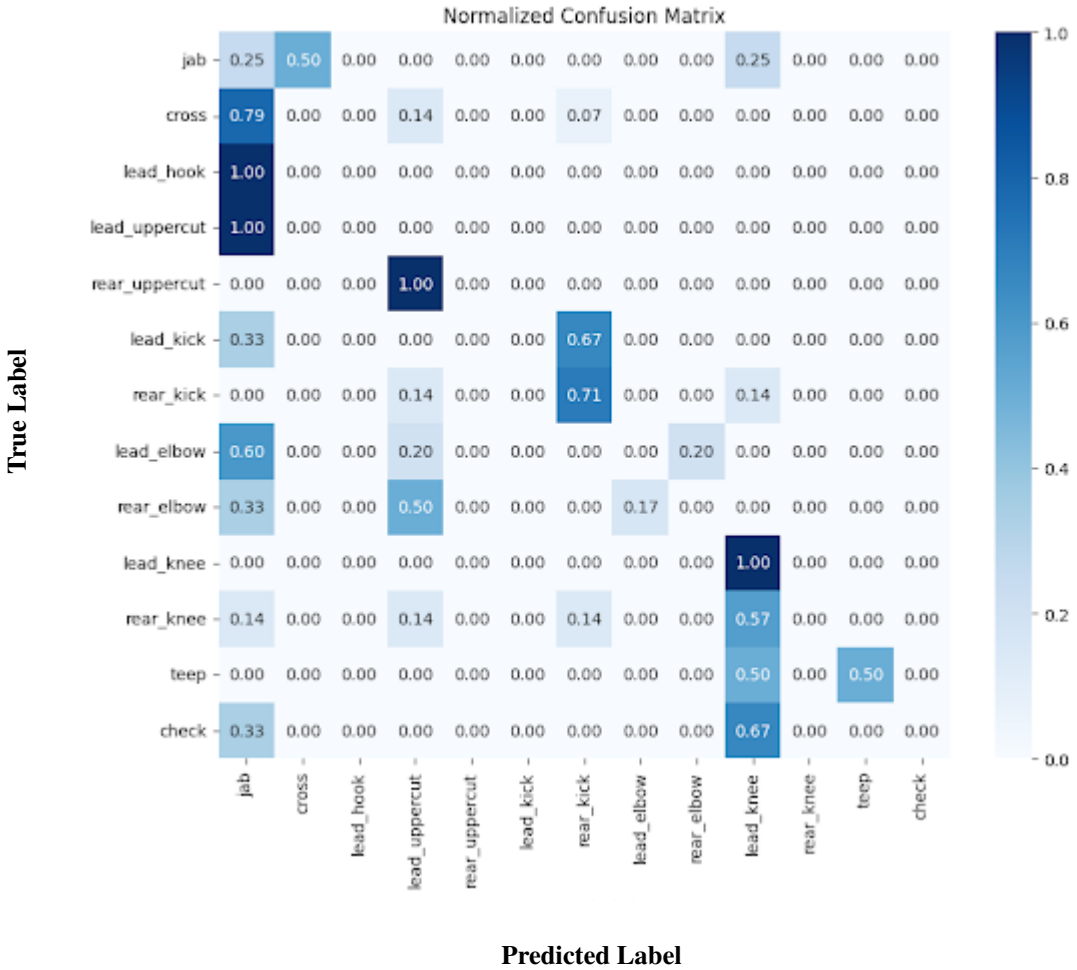


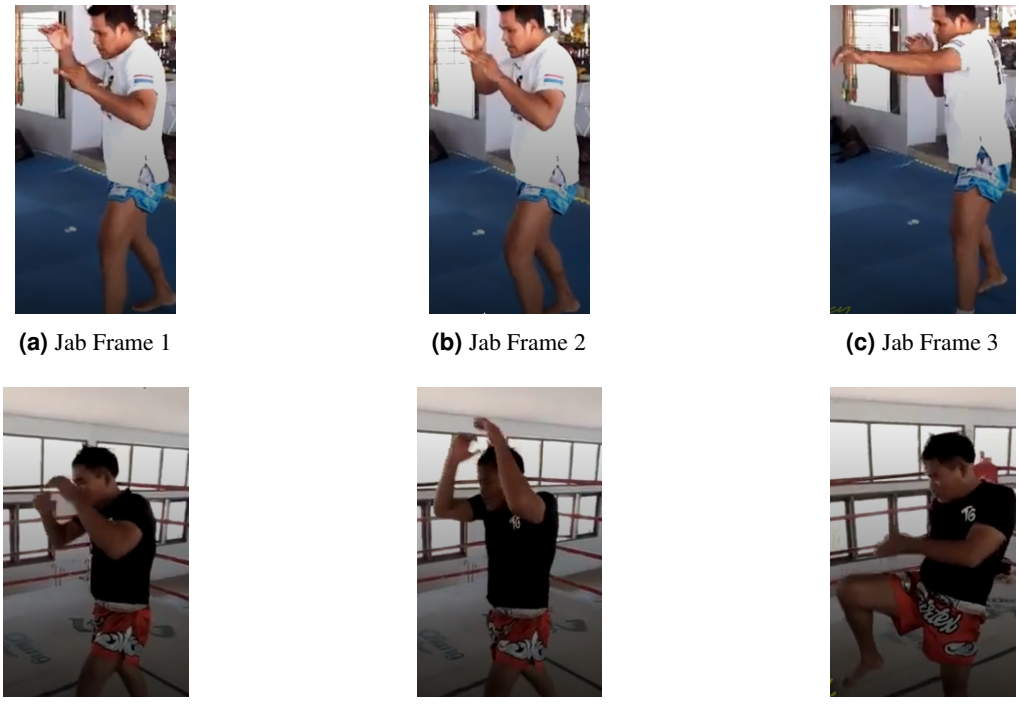
Figure 3. Confusion Matrix from *VideoMAE* demonstrating frequent misclassification between similar strike types.

RESULTS

Overview

Our results demonstrated significant advancements in the domain of Muay Thai action recognition, exceeding prior benchmarks set by *StrikeMetrics* [9], a prototype tool for combat sport analysis that leverages Computer Vision and Machine Learning techniques. *StrikeMetrics* [9] achieved an accuracy of 12.5%, it operated with only two labels, compared to the 14 and 8 labels used in our model configurations. This distinction highlights not only the complexity of our task but also the robustness of our approach in handling fine-grained classifications across a wider set of strikes.

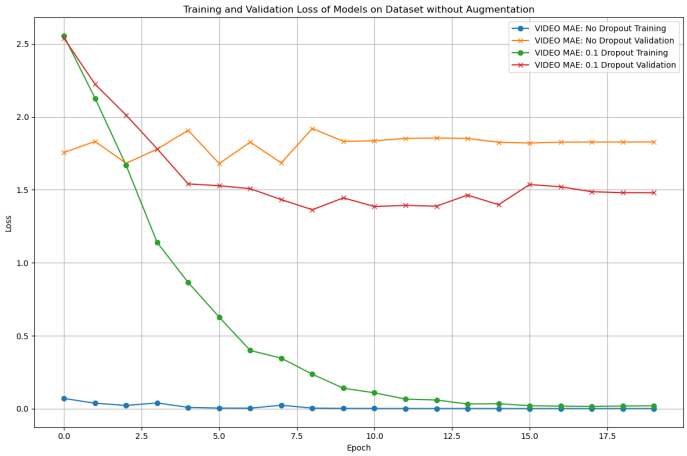
A major contributor to our success was the dataset creation pipeline, which allowed us to efficiently label and preprocess clips. Initially, we hand-labeled 21 clips to begin the project, but after transitioning away from *MMAction2* [7] and PKL-based inputs, we developed a scalable pipeline that transformed JSON timestamp files into labeled MP4 clips. This innovation enabled us to expand the dataset to 410 clips, later augmented to 1,640 clips, providing ample data for training and validation.



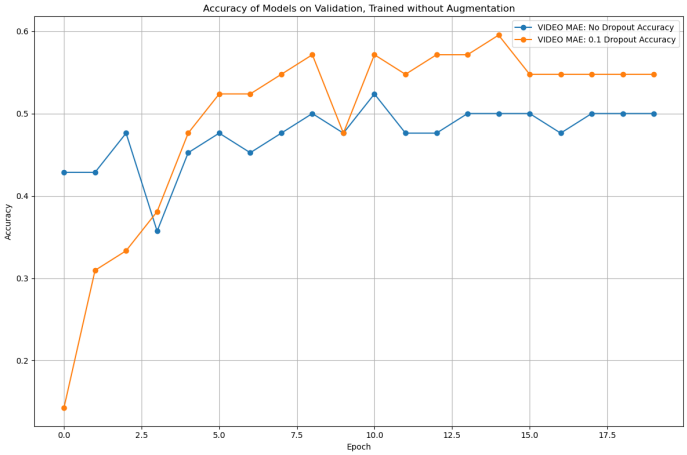
(a) Jab Frame 1 **(b)** Jab Frame 2 **(c)** Jab Frame 3

(d) Knee Frame 1 **(e)** Knee Frame 2 **(f)** Knee Frame 3

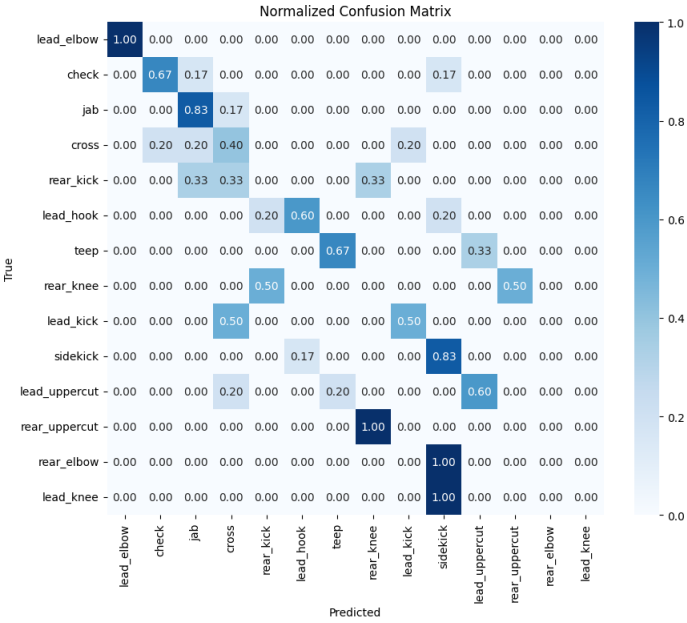
Figure 4. Frame Sampling Example of Misclassified Strikes



(a) Epochs vs. Validation Loss - Base Dataset

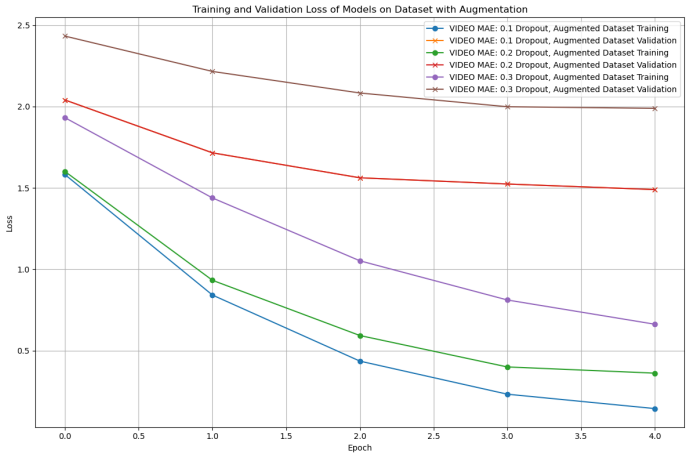


(b) Epochs vs. Model Accuracy - Base Dataset

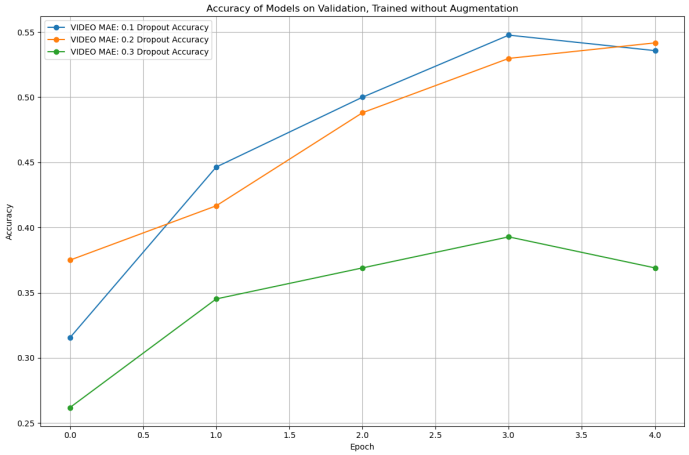


(c) Predicted vs. True - Original Dataset Confusion Matrix (0.1 Dropout)

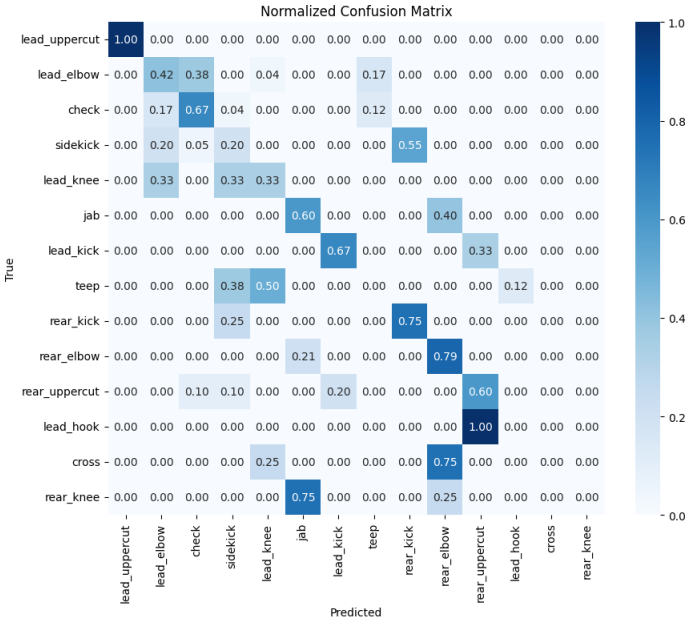
Figure 5. Model Results with Base Dataset. Subfigures show (a) Training vs. Validation Loss, (b) Model Accuracy, and (c) the Confusion Matrix.



(a) Epochs vs. Validation Loss - Augmented Dataset



(b) Epochs vs. Model Accuracy - Augmented Dataset



(c) Predicted vs. True - Augmented Dataset Confusion Matrix(0.1 Dropout)

Figure 6. Model Results with Augmented Dataset. Subfigures show (a) Training vs. Validation Loss, (b) Model Accuracy, and (c) the Confusion Matrix. The augmented dataset significantly improved model robustness and generalization.

Video MAE

Video MAE was trained on two versions of the dataset, one without augmentations and one with augmentations. The discussion will be split as such. Our initial model proved to perform poorly despite our small dataset size. We achieved a maximum training accuracy of 52% on our base model without dropout and were able to gain a modest 5% validation increase by including a 0.1% dropout layer. This model was trained for 20 epochs, and showed that this model achieves an optimal loss on training and validation in less than 10 epochs as after that validation loss stagnates or in some cases increases as shown in Figure 5(a). These results allowed us to tailor later training to use fewer epochs while maintaining accuracy. However, these models were less comparable on a separate testing dataset. The 0.1% dropout model outperformed the No Dropout model by 12% with a 56% accuracy compared to 44% 3(b). Highlighting that even with these poor results overfitting was occuring in the non-dropout model. In addition to discovering the need for dropout, we also highlighted misclassification errors utilizing confusion matrices shown in Figure 5(c). The confusion matrix showed that the models had issues classifying strikes on different side of the body. An example of this is confusion between the jab and the cross. Even with the jab being our most common strike in the dataset we were only able to achieve an 83% success rate in classification. The classification errors occurred on the cross which is a straight strike that happens on the other side of the body. These errors are seen throughout the confusion matrix highlighting our model's inability to differentiate the side from which a strike originates.(Figure 5(c)). These results were attributed to dataset size with the belief that data augmentation would help to address these errors. This model scored 7% on our inference pipeline of an annotated video not included in our testing sets.

The model trained on the augmented larger dataset however proved the contrary, as Accuracy remained the same or worse with slightly different model architectures and even longer training times as a result of the increase in dataset size. Our limit was 5 epochs because even then it took 3+ hours, but we confirmed that a 0.1% dropout rate was optimal as it achieved the lowest validation loss out of the set of tested models as seen in Figure 4(a). In addition to that it scored a similar accuracy to our previous models at 54% on training and 52% on validation. However, when tested on our inference pipeline it achieved similar scores to the previous models of 7%.

In comparing these models we see that our data augmentation did improve our training effectiveness even if accuracy scores remained relatively the same. This is most clearly seen in accuracy improvements over epoch, with augmented data allowing scores of over 50% in just 2 epochs compared to the non-augmented datasets taking 4 or more epochs to reach similar scores. The latter models also resulted in much more readable confusion matrices allowing us to more easily diagnose strike classifications. Figure 4(c) includes a confusion matrix from this batch of testing and we identify that our problem strikes are punches getting confused with elbows, and a failure to correctly identify knees. These realizations led to the decision to flatten our dataset in hope that increased representation of simpler classes would aid classification. Furthermore, it highlighted issues with the annotation method particularly the definition of a where a strike begins. Shown below are frames taken from the beginning of a knee and frames from a jab in Figure 4(C). In this example we see that for the majority of the frames in a, b, d, and e that the hand position is extremely similar with the strikes only significantly differing in the final frame which is where the strike fits our definitions. These images highlight the similarity for many frames of each strike up until the end where they are distinct. These strike timestamps in addition to the uniform sampling methodology used for model input results in frames being pulled equally from all parts of the clip. This likely resulted in the majority of the input between confused strikes like rear_knee and jab being very similar with the only differences coming in the last input frame or so. These highly similar inputs could be the reason the model had a hard time differentiating between some of these strikes in addition to dataset limitations. In conclusion the Video MAE model helped to influence data augmentation, and flattening and provided a point of comparison for judging other model architecture's effectiveness.

TimeSformer

By employing the *TimeSformer* [2] model (pretrained on *Kinetics-400* [5]) with focal loss to address class imbalances, we achieved a final testing accuracy of 70.18% on unseen data (5.6 $\times$ benchmark). This improvement was bolstered by augmentations like cropping, flipping, and temporal adjustments, which enhanced generalization. Flattening the label set further reduced confusion between similar strikes (e.g., jabs and crosses) by grouping them into categories like "straight punches" and "kicks."

These results not only underscore the advantages of our methodology but also mark a significant

improvement over the previous state-of-the-art in this domain. By incorporating a detailed label taxonomy, an efficient dataset pipeline, and advanced augmentation techniques, our approach sets a robust benchmark for future combat sport action recognition tools.

Impact of Augmentation and Label Flattening

We conducted experiments with augmented and non-augmented datasets, as well as original and flattened label sets. The results from the *TimeSformer* model are summarized in Table 1. Augmentation significantly improved accuracy, while label flattening reduced confusion between similar strikes.

Configuration	Inference Accuracy (%)	Observations
Original Labels, No Augmentation	29.82	Baseline performance
Original Labels, With Augmentation	33.33	Improved generalization
Flattened Labels, No Augmentation	68.42	Reduced label confusion
Flattened Labels, With Augmentation	70.18	Best overall performance

Table 1. Comparison of testing accuracy for different configurations for *TimeSformer*.

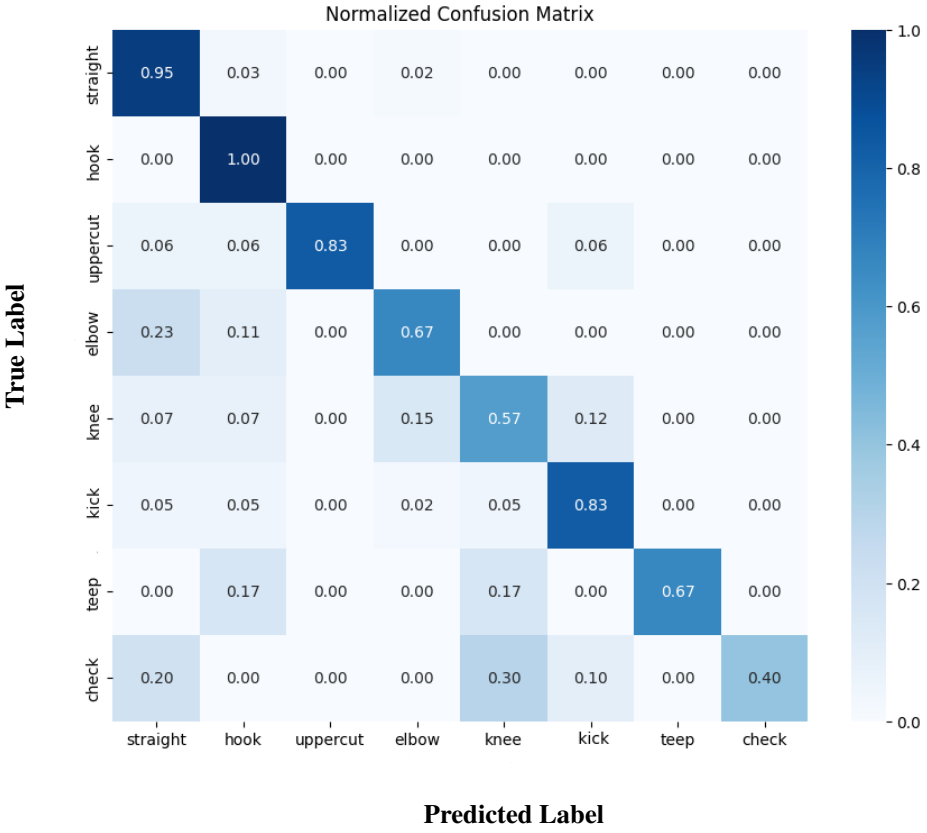


Figure 7. Confusion Matrix from final model showing improved generalization on similar strike types.

Confusion Matrix for Final Model

Figure 7 illustrates the confusion matrix for the best-performing configuration. This visualization highlights the significant reduction in confusion for the flattened label set.

LIMITATIONS

Dataset Size and Label Representation

One of the main limitations of this work was the dataset, particularly its size, the quality of the raw data, and the annotation strategy. Even though we aimed to create a new dataset tailor-made for this task, it was

still relatively small in size even after efforts to supplement it. Another major issue was that many strikes, before label flattening, had similar movements, often leading to misclassification.

For example, both lead and rear knee strikes have virtually identical setups and means of execution, the only difference being the leg used. Another example is that elbow strikes are usually preceded by a "hand trap," which visually resembles straight punches such as jabs or crosses. This visual similarity often caused elbow strikes to be incorrectly classified as punches.

These issues are reflected in the confusion matrices produced during experimentation, which singled out probable causes of model misclassification. Increasing the size of the dataset would allow models to build more specific representations of each strike, potentially enabling higher accuracy on an unflattened label set. Furthermore, many strikes include preliminary sequences that are similar to other strikes, further complicating classification.

Strikes: Asymmetry and Heterogeneity

The dataset also suffered from a limited diversity of strikes and imbalanced strike representation. In Muay Thai, as in other striking disciplines, lead-side strikes—jabs, left hooks—are thrown with a higher frequency than their rear or "power-side" counterparts. Similarly, some strikes, such as uppercuts, are less frequent in a fight because they are situational and usually met with specific defensive postures from the opponent.

Addressing this imbalance would require additional data collection to cover a wider range of Muay Thai-specific situations or augmented strategies that artificially create balance by generating more samples of underrepresented strikes.

Annotation Challenges

Annotation was also a significant factor. The footage was annotated by non-expert practitioners, which introduced variability into the definition of where strikes begin and end. Our goal of consistent definition is likely contaminated by variations introduced in the labeling and reinforced by the small size of the dataset.

The annotation method itself could also be improved by revisiting skeleton-based keypoint data or bounding box annotations. This additional information might help the model to better identify strike mechanics and develop more accurate representations of action classes.

Raw Data Limitations

The raw data presented a lot of challenges, mainly because of the variation of camera angles and cases of occlusion. Many of the videos acquired through the available datasets featured diverse views that sometimes occluded strikes or clipped parts of the subject's body. In effect, some of the samples of strikes had fewer visual details than others, thus compromising the classification accuracy.

This would, therefore, mean that the model had to generalize its representations from these inconsistent camera angles. Allowing many recording angles, as was done in the Martial Arts Dance Sports dataset, would have been an improvement in and of itself. Similarly, the approach followed in *Jabbr* [1], that of gathering martial arts data from multiple angles, gives a basic framework for improving generalization in future versions.

Testing the model in different conditions—for example, with different camera angles and partially occluded views—is another area for improvement. The current validation relied on a fixed camera angle, which limited the ability to test the effectiveness of the model in practical situations. More extensive testing, including a variety of perspectives, may provide a more holistic view of the model's robustness.

Model Limitations

A critical limitation of the models used was that they relied on timestamped video segments, which required pre-segmented data to perform the classification. This dependence limited the inference pipeline and prohibited classifying data in continuous streams. A more robust approach would involve updating the model to handle continuous streams of data and to output action probabilities in real time.

Moreover, a major bottleneck was the required computational power for training: for instance, *VideoMAE* took several hours to train on an A100 GPU, even though the dataset size is small. Increasing the dataset size would make this problem worse. In order to tackle this, more efficient training strategies can be adopted, which optimize batch sizes, make use of mixed precision training, and test alternative

optimizers. Tools like ‘torch.compile‘ can also be used to further reduce the training times and improve the scalability.

Recommendations and Future Directions

It would help to add more diverse and balanced examples to the dataset, increasing the model's performance. It could also be obtained by gathering new footage from Muay Thai sparring sessions with a better representation of the underused strikes, including additional viewpoints. Engaging experienced practitioners for annotation would further enhance dataset quality.

The models supporting continuous classification of video streams would make the usability and adaptability of the system much greater. Further, computational efficiency could be improved through mixed precision training and other methodologies, which would lead to faster iterations and better scalability. Research into ensemble techniques or integration of pose-based features with video data may also improve the accuracy of classifications and generalize them.

More detailed assessment of the model's robustness would be achieved by the evaluation across different camera angles and conditions. Integration of multi-angle footage into both training and validation could, in a more faithful way, reproduce real-world situations and confirm the model's utility in a variety of applications.

CONCLUSION

At the start of this project, we identified a significant gap in data specific to Muay Thai and the lack of models capable of predicting complex strike labels beyond basic kicks and punches. In response, we developed an efficient dataset creation pipeline that incorporates augmentation and flattened labeling tailored to Muay Thai. Transitioning from our initial skeleton-based pose estimation model using MMAction2[7], our work demonstrated notable advancements in Muay Thai action recognition, with both *VideoMAE* [11]and *TimeSformer*[2] models surpassing the benchmark set by StrikeMetrics[9].

In the future, we aim to expand our project by increasing the dataset to include a wider variety of strikes and diverse recording angles. We will also incorporate expert practitioner labeling to ensure balanced strike representation and create a dataset that generalizes effectively to real-world scenarios. Additionally, we plan to update our model and dataset pipeline to process continuous data streams for real-time predictions and implement more efficient training strategies. These improvements will align the model with our proposed fight performance feedback application, reduce reliance on time-stamped data, and enhance scalability. We also plan to explore multimodal large language models, such as LLaVA-NeXT-Video [14]—a vision-text-to-text chatbot model designed to integrate video and image data for a more comprehensive understanding of video inputs.

INDIVIDUAL CONTRIBUTIONS

Teddy Danielson

Teddy Danielson contributed to many different parts of the project and supplied a great deal of valuable input in both data preparation and model development. One contribution Teddy made was creating the timestamp labeling procedure, which served as the foundation for how video footage was segmented into labeled portions. This process ensured the rigorous annotation of the start and end times of strikes, which enabled the creation of a structured dataset ready for model training. Moreover, Teddy was involved at the early stages of this project, having labeled a part of the dataset—something that required an enormous review of fight footage in order to achieve uniformity and accuracy.

Teddy Danielson also brought unique expertise both as a Muay Thai fighter and trainer, so he could offer very important insights regarding the different types of strikes, their mechanics, and patterns. His knowledge of Muay Thai guaranteed not only the technical accuracy of the project but also significantly raised the ability of the team to annotate and analyze the data.

Then came the important decision to flatten the label set, whereby strike classifications were simplified because similar strikes were often misclassified—a decision made by Teddy as a result of his experience in the field. This balanced the complexity of the dataset with the practical capability of the model and drastically improved its performance. The way Teddy could combine technical insights with his experience in martial arts had a big impact on the overall success of the project.

On the technical front, Teddy developed the script for the skeleton data of MediaPipe at the beginning of the project. This script was instrumental in the investigation of pose-based action recognition, providing key insights that helped drive the transition to video-based models. Teddy then took the lead in applying and optimizing the *TimeSformer* model: this involved modifying the pretrained *TimeSformer* for the classification of Muay Thai strikes, re-calibrating hyperparameters, and testing methods like focal loss to counteract class imbalances. Being the lead in model optimization, Teddy's work helped us to attain the highest accuracy in testing—70.18% for this project.

Furthermore, Teddy designed and implemented the data augmentation process, an integral part in increasing the robustness of the dataset. It increased the dataset from 410 to 1,640 clips by operations including cropping, flipping, temporal adjustments, and adding grain, thus enabling the model to generalize better on unseen data.

Regarding the report, Teddy contributed to methodology and results sections, primarily focusing on insight involving the *TimeSformer* model, data treatment, and label flattening. Teddy also formatted the report in LaTeX and helped group members add visuals to their sections in order to contribute to a professional and readable final paper.

Anthony Wong

Anthony Wong's contributions involved data sourcing, annotation, pre-processing, and VIDEO MAE implementation, testing, and optimization. He utilized his 4+ years of Striking and Muay Thai experience to help develop the strike classifications and annotation criteria for how timestamps would be implemented. Anthony utilized his expertise to identify existing datasets which could be useful such as HMDB51, UCF101 and MADS, along with discovering expert resources like Sylvie Von Douglas Ittu's footage of professional fighters demonstrating techniques and shadow boxing. In addition to dataset identification and raw data collection,

Anthony Wong identified the VIA annotation tool, and utilized it to annotate all of the sourced Database footage into the projects specific strike definitions, as well as performing the bulk of dataset annotation on expert videos. He expanded the dataset from 20 to around 400 clips annotating 20+ full length videos and 100s of existing dataset clips. Anthony also implemented a pre-processing notebook to convert videos and their annotations into the UCF dataset format setting it up to be easily augmented later by Teddy Danielson.

In addition to dataset tasks Anthony Wong also researched, implemented and fine tuned the *VideoMae* model which served as a datapoint for model comparisons and goals which eventually led us to the final *TimeSformer* Model. With regards to the report, Anthony Wong contributed mainly to the methodology sections describing the data collection and annotation process, in addition to the VIDEO-MAE results and methodology and the limitations.

Arushi Tyagi

Arushi Tyagi focused on model development, optimization, and dataset handling. She implemented the posec3d_ucf101 pretrained model, tailoring it to the unique challenges of Muay Thai action classification. Her work involved extensive experimentation with hyperparameters, fine-tuning, and testing to ensure the highest possible accuracy and robustness of the model.

One of Arushi's most impactful contributions was the integration of pose-based action recognition into the project pipeline. She utilized MMAction2's pose estimation tools to extract skeletal keypoints, enabling precise identification of strikes and movements. Arushi also implemented and optimized the posec3d_ucf101 model which was used in the early stages, fine-tuning it for the nuances of Muay Thai techniques.

Beyond technical implementation, Arushi contributed insights to the limitations sections of the project report.

Loni Halsted-Ruelas

Loni Halsted-Ruelas took on a leadership role, scheduling and facilitating the group biweekly in-person and Zoom meetings. She kept a system of meeting notes and to-do lists per meeting. She also provided reminders regarding upcoming deadlines and responsibilities for the project and coordinated the check-ins with the TAs.

On the technical side, she played an important role in building the hand-labeled dataset. She labeled the data from the YouTube-sourced videos and part of the self-recorded video datasets, and also in

converting these labels into JSON format. Also, she conducted research to provide an overview of existing martial arts datasets and models which helped to better inform the scope of the project. She also contributed to the research and implementation of Mediapipe for the skeleton-based data labeling.

Loni also collaborated on various project elements, including the project proposal, class presentation slides, final report, and materials shared with the TA. She specifically contributed to the conclusion section of the final report.

Rintaro Oshima

Rintaro Oshima focused on researching and discussing methodology. He researched about many kinds of methodology which may help our project including CNN, GCN, LSTM,and attention mechanism, and summarized each pros and cons. In methodology, he visualized our project path.

Rintaro contributed insights to the introduction sections of the project report.

BIBLIOGRAPHY

[1] AI, J. (2024). Jabbr: Ai-powered martial arts analysis. https://jabbr.ai/. Accessed: Dec. 9, 2024.

[2] Bertasius, G., Wang, H., and Torresani, L. (2021). Timesformer: Space-time attention for video understanding. *arXiv preprint arXiv:2102.05095*.

[3] Dutta, A. and Zisserman, A. (2019). The VIA annotation software for images, audio and video. In *Proceedings of the 27th ACM International Conference on Multimedia*, MM '19, New York, NY, USA. ACM.

[4] Images, G. (2024). Royalty-free stock photos, illustrations, and videos. https://www.gettyimages.com. Available at https://www.gettyimages.com.

[5] Kay, W., Carreira, J., Simonyan, K., Zhang, B., Hillier, C., Vijayanarasimhan, S., Viola, F., Green, T., Back, T., Natsev, P., Suleyman, M., and Zisserman, A. (2017). The kinetics human action video dataset. *arXiv preprint arXiv:1705.06950*.

[6] Kuehne, H., Jhuang, H., Garrote, E., Poggio, T., and Serre, T. (2011). HMDB: a large video database for human motion recognition. In *Proceedings of the International Conference on Computer Vision (ICCV)*.

[7] of MMAction2, C. (2021). Mmaction2: Openmmlab's next generation video understanding toolbox and benchmark. GitHub repository. Accessed: Oct. 16, 2024.

[8] Pang, Y., Wang, Y., Wang, Q., Li, F., Zhang, C., and Ding, C. (2024). Applications of ai in martial arts: A survey. *Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology*.

[9] Sippel, S., Cherinet, J., and Mostert, C. (2024). Github - sswhitehat/strikemetrics—kickboxing-ai-tool: Boxing stat tracker. GitHub repository. Accessed: Oct. 16, 2024.

[10] Soomro, K., Zamir, A. R., and Shah, M. (2012). Ucf101: A dataset of 101 human actions classes from videos in the wild. Technical Report CRCV-TR-12-01, Center for Research in Computer Vision (CRCV), University of Central Florida. Accessed: 2024-12-09.

[11] Tong, Z., Song, Y., Wang, J., Wang, L., Lu, Y., and Dai, B. (2022). Videomae: Masked autoencoders are data-efficient learners for self-supervised video pre-training. *arXiv preprint arXiv:2203.12602*.

[12] von Duuglas-Ittu, S. (2024). Sylvie von duuglas-ittu - muay thai resources. https://www.youtube.com/@8limbsUs. Accessed: 2024-10-12.

[13] Zhang, F., Bazarevsky, V., Vakunov, A., Tkachenka, A., Sung, G., and Grundmann, M. (2020). Mediapipe hands: On-device real-time hand tracking. *arXiv preprint arXiv:2006.10214*.

[14] Zhang, Y., Li, B., Liu, h., Lee, Y. j., Gui, L., Fu, D., Feng, J., Liu, Z., and Li, C. (2024). Llava-next: A strong zero-shot video understanding model.