ACHIEVING DEXTEROUS MANIPULATION VIA VISION FOR STROKE PATIENTS

Vishesh P Arora, Rosh Ho

ABSTRACT

This paper presents a novel approach to automatic grasp selection for active hand exoskeletons and prosthetic hands that utilize depth maps in addition to RGB images. Prior research has only correlated objects to grasps using RGB data with Convolutional Neural Networks (CNN). However, depth estimation has improved significantly and unique depth-based features of objects can add valuable information on grasp selection. We used the labeled datasets from DeepGrasping and ImageNet for training and the HandCam dataset for testing, achieving improvements in grasp classification accuracy and more robust grasp selection despite imbalanced training datasets. The results highlight the potential of deep learning to improve dexterous manipulation. Code is available at `https://github.com/wish2023/Dexterous-manipulation-via-vision` and `https://github.com/roshho/tempDL`. Data is available from [1]

1. INTRODUCTION

Of the 25% of population that experiences stroke, 50-80% experience upper body extremity impairment in acute phase and 40-50% of patients continue to experience upper body immobility in sub-acute phase. Despite improved affordability from soft robotics applied to stroke rehabilitation, 75% of caregivers [2] and patients who experienced greater spinal cord deterioration are still unable to leverage rehabilitation to improve conditions, indicating a need for non-invasive orthotics.

2. RELATED WORK

Recent research [2] [3] [4] in the field have demonstrated interest and potential from interpreting EMG signal; however the studies are either limited in accuracy or limited to interpreting for two or three grasps instead of the commonly used five or six [5]. Combinations of alternate readings, such as EEG [3], has shown great promise, though possibly less practical. This paper seeks to explore employing computer vision in place of EMG and EEG, identifying the efficacy of mapping images and depth maps of an array of objects to 5 commonly used daily grasps: Key, Pinch, Power, Three Jaw Chuck, and Tool.

3. DATASETS

The datasets used in this study build upon prior work by [1] and are tailored for the evaluation of grasp recognition:

3.1. DeepGrasping Dataset

The DeepGrasping dataset [6] includes 1,035 images of 280 objects, each with a resolution of 640×480 pixels. Objects were annotated with one of five grasps based on their most natural fit. A significant bias exists in the dataset, as the power grasp dominated, while key and tool grasps were hardly represented. This bias motivated the creation of more balanced datasets.

3.2. ImageNet-derived Dataset

Due to the lack of representation from the key and tool grip in the DeepGrasping dataset, a dataset was curated from ImageNet [7], containing 5,180 images grouped into 25 object categories, such as balls, scissors, and utensils. Each image was labeled with the most appropriate grasp, emphasizing balanced representation across all grasp types. This dataset fills the gaps in the representation of key and tool grasps. The power grasp still remains a dominant class.

3.3. HandCam Testing Dataset

The HandCam dataset was specifically created by [1] for testing. This dataset comprises of 250 images of 50 objects, captured using a prosthetic hand camera. Unlike the other datasets, each grasp type is uniformly represented, with ten objects chosen per grasp and photographed from five different perspectives. This dataset is exclusively used for evaluation, ensuring the model is tested on unseen, out-of-distribution data.

Table 1 highlights the percentage distribution of grasps across all datasets. The DeepGrasping and ImageNet datasets were used to train the classification model, while the HandCam dataset was used for testing.

4. EXTRACTING DEPTH MAPS

Depth maps were generated using [8], a monocular depth estimation model. This method was trained in both a supervised and unsupervised fashion. The approach incorporated data

	DeepGrasping	ImageNet	HandCam
Train/Test	Train	Train	Test
Grasp			
Key	0.0 %	11.8 %	20.0 %
Pinch	21.8 %	10.6 %	20.0 %
Power	47.0 %	47.5 %	20.0 %
Three Jaw Chuck	28.0 %	19.2 %	20.0 %
Tool	3.2 %	10.9 %	20.0 %

Table 1. Distribution of training and test data. The power grasp is a dominant class in the training set, with 47.5% of samples having a power label

augmentation techniques to obtain more robust features. The approach also extracts priors from pre-trained encoders to further boost model understanding, make it a reliable framework to use for our downstream task of hand grasp classification.

5. MODEL ARCHITECTURES

We created three models for our experiments: the RGB ResNet50 baseline, the extended RGB-D ResNet50, and the ResNet50 + Depth Fusion model:.

5.1. RGB ResNet50

ResNet50 is a pretrained model used for image classification tasks of RGB images [9]. It consists of a a series of convolutions and residual connections. These act as skip connections to mitigate the vanishing gradient problem, allowing for better training in earlier layers of the network. The classification head of ResNet50 was modified to predict between the 5 classes of various grips ("3 jaw chuck," "key," "pinch," "power," and "tool").

5.2. RGB ViT-B-16

ViT-B-16 is a pre-trained transformer model made with 12 attention heads, 768 hidden dimensions,, 12 transformer layers, and 86M paramters. While this model doesn't have the largest pixel patch size, it function as a good compromise for compute power and good performance. This has been adapted to assign 5 types of grasps ("3 jaw chuck," "key," "pinch," "power," and "tool"), trained on DeGol's labeled data.

5.3. RGB-D ResNet50

To incorporate depth information, the ResNet50 model was modified to accept four-channel inputs (RGB + Depth). This was done by adjusting the input channels to four in the first convolutional layer. The weights of the first 3 channels were frozen to the default RGB weights, while the fourth channel weights were randomly initialized. The remaining architecture is identical to the original ResNet50, with the classification head adapted with 5 output neurons.

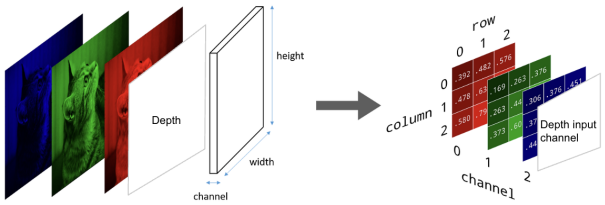

Fig. 1. RGB-D ResNet50 input kernel: A new input channel for depth is added. The RGB channels maintain their pre-trained weights before finetuning, while the weights of the input depth channel are randomly initialized. Kernel diagram created by [10].

5.4. RGB-D ViT

Similar to RGB-D ResNet50, an additional channel was added to account for depth channel to train RGB + depth data. In this context, the most standard ViT with no pre-trained weights was used in hopes to have the best efficacy with no affect from weights trained on RGB data.

The ViT architecture also uses 16x16 patches with 12 attention heads.

5.5. ResNet50 + Depth Fusion

This approach was designed to obtain depth-specific features, similar to how ResNet50 extracts RGB specific features. The RGB channels are passed through the original ResNet50 backbone, while the depth data is processed by an autoencoder network to extract high-level features. Before classification, the encoded depth features are concatenated with the flattened ResNet50 features before being passed to a classification head.

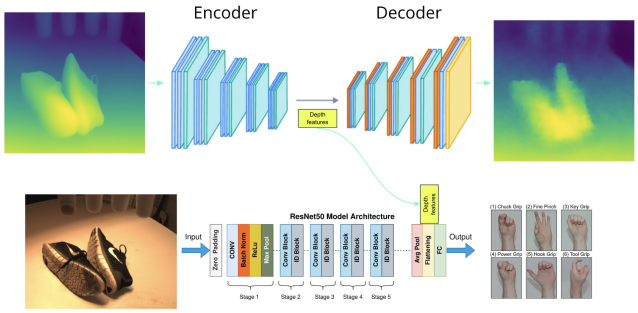

Fig. 2. Depth Fusion: The autoencoder (above) is trained to reconstruct the depth map, which act as pre-trained depth features. The encoder is utilized as a backbone to extract depth features. Concurrently, the ResNet50 backbone (below) processes RGB channels. RGB and depth features are then concatenated before being passed to FC (classification head). Autoencoder diagram by [11], ResNet diagram by [12]

5.5.1. *Depth Autoencoder*

The depth autoencoder consists of an encoder and a decoder. The encoder applies convolutionals to the input to extract a compact and high-level representation of the depth map. The decoder attempts to reconstruct the original depth map from the bottleneck layer (the encoded features) using deconvolutions. During training, the network aims to minimize the mean squared error between the reconstructed image, and the input image. Once the autoencoder is trained to successfully reconstruct depth maps, the encoder is used as a depth feature extractor, generating a fixed-length feature vector (length 512) for fusion with the RGB features from the ResNet backbone.

6. TRAINING DETAILS

The CNN models were trained using a single Nvidia Tesla T4 on the following hyperparameters:

- **Number of Epochs:** 100.
- **Batch Size:** 64.
- **Learning Rate:** 1×10^{-3}.
- **Optimizer:** Adam optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$.
- **Loss Function:** Cross-entropy loss.

The ViT models were trained using an Nvidia RTX3060 Laptop GPU and on the following hyperparameters:

- **Number of Epochs:** 10.
- **Batch Size:** 32.
- **Learning Rate:** 1×10^{-3}.
- **Optimizer:** Adam optimizer weight decay: 0.05.
- **Loss Function:** Cross-entropy loss.

7. RESULTS

Our results are summarized in Table 2.

Model	Accuracy (%)
RGB ResNet50 (Baseline)	79.6
RGB ViT-B-16 (Baseline)	46.8
RGB-D ResNet50	62.8
RGB-D ViT-B-16	38.8
Depth Fusion	**82.4**

Table 2. Classification accuracy of different models.

The RGB baseline had an accuracy of 79.6%. The accuracy of the RGB-D ResNet50 was 62.8%, indicating that adding a depth channel to ResNet hinders learning as the RGB and depth features did not synergize well. This was also the case with using transformers, where the accuracy dropped from 46.8% to 38.8%. The sub-optimal performance found in the ViT models are likely a result of a small training sample. The inferior performance found in the RGB-D ViT model has been congruent with existing literature stating worse performance than CNN when analyzing solely depth data [13]. Extracting RGB and depth features independently was the most successful, as it outperformed the baseline with an accuracy of 82.4%, indicative of robust feature extraction across all channels.

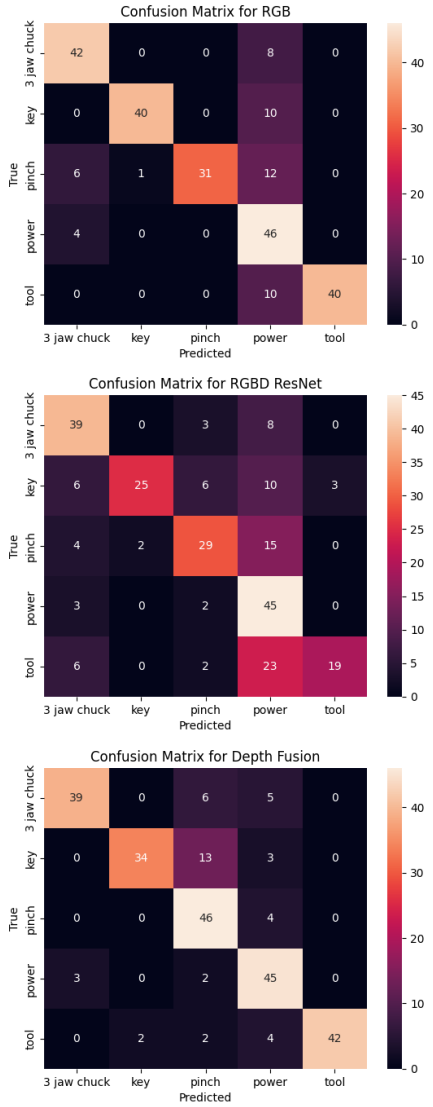

Fig. 3. Confusion Matrices of the RGB ResNet50 + Depth Fusion model

Upon examining the confusion matrix of the RGB baseline architecture in Figure 3, it is expected that the model has lower precision for the "Power" class. As mentioned in Ta-

ble 1, this is a dominant class in the training set, making the model more prone to predicting this class. This is also evident in the RGB-D ResNet confusion matrix, as the model is unable to obtain good representations of depth maps, leading to even more misclassifications of grasps. However, the precision for the power class significantly improves for the depth fusion model, which is evident from the confusion matrix. The depth maps provide robust information that cannot be obtained from RGB channels regarding power grips that are not only representative but also unique to this class. The depth fusion model is most confused between key and pinch grasps. Objects that require these grasps, such as pencils, cards, or keys, tend to have relatively flat depth maps. As a result, depth maps do not offer sufficient distinguishing information, leading to an increased misclassification between these two grasp types.

8. FAILURE CASES

The best performing model (RGB ResNet50 + depth fusion) has surpassed the baseline with an accuracy of 82.4% on a uniformly distributed test. However, there are some cases that cause this model to make incorrect predictions. As evidenced by figure 3 and depicted by figure 4, the classifier is easily confused between key and pinch grasps. This is likely due to the flatness of their depth maps; it is challenging to identify unique depth related features from a scattered pile of papers on a flat surface.

The model also wrongly classifies the grip of a wine glass to be that of a pinch, rather than a 3 jaw chuck. The cylindrical nature of the glass hold closely resembles the depth features of objects typically held with a pinch grip, such as batteries.

Lastly, there are a few instances of the model incorrectly predicting that an object correlating to a tool grasp should be assigned a power grip. A common pattern to this misclassification is when the model is exposed to a relatively bulky object.

9. DISCUSSION

Adding depth in a separate channel, in addition to RGB data, for mapping grasps as proven to be effective. Further tests with depth-specific CNN and ViTs need to be conducted to be find a more optimal model for the dedicated depth fusion analysis. These results has been congruent with past research in the prosthetic field, with objectives aiming to improve object grasping based on various combinations of multi-modal inputs [14] [15].

The primary drive for greater improvement is from a larger dataset with data labeling in a more naturalistic, intuitive manner. This is contrary to DeGol's method, requiring participants to label grasps to object based on mental visualization as opposed to recording grasps with motion capture or flex sensors.

Fig. 4. Some misclassified grasps of the RGB ResNet50 + Depth Fusion model

10. CONCLUSION

In this project, we explored the effectiveness of combining depth maps with RGB images to perform automatic grasp selection in prosthetic hands. Our best-performing model was the ResNet50 + Depth Fusion that achieved an accuracy of 82.4% on a uniformly distributed test set, surpassing the base-

line accuracy of 79.6%. However, failure cases revealed specific challenges with the model, such as the flatness in depth maps for certain grasp types (key vs. pinch grasps) and misclassifications influenced by object shape and size.

This highlights the importance of extracting richer depth features and perhaps even integrating more modalities to improve the model's ability to distinguish similar grasps across similar objects. Future work could focus on further exploring vision transformers and expanding the dataset to better represent certain grasps, improving the applicability for prosthetic hand control.

11. AUTHOR CONTRIBUTIONS

Vishesh's contributions:

- Extracted depth maps, and preprocessed RGB-D data to input into all CNN models.
- Trained, benchmarked, and conducted error analysis on the following models: RGB ResNet50, RGB-D ResNet50, ResNet50 with Depth Fusion.

Rosh's contributions:

- Research on status quo research progress, alterantive solutions from tangential fields, and proposed direction.
- Trained, benchmarked, and conducted error analysis on the following models: RGB ViT-B-16, RGB-D ViT

12. REFERENCES

[1] Joseph DeGol, Aadeel Akhtar, Bhargava Manja, and Tim Bretl, "Automatic grasp selection using a camera in a hand prosthesis," in *EMBC*, 2016.

[2] Eric C. Meyers, David Gabrieli, Nick Tacca, Lauren Wengerd, Michael Darrow, Bryan R. Schlink, Ian Baumgart, and David A. Friedenberg, "Decoding hand and wrist movement intention from chronic stroke survivors with hemiparesis using a user-friendly, wearable emg-based neural interface," *Journal of NeuroEngineering and Rehabilitation*, vol. 21, no. 1, Jan 2024.

[3] Haoyang Li, Hongfei Ji, Jian Yu, Jie Li, Lingjing Jin, Lingyu Liu, Zhongfei Bai, and Chen Ye, "A sequential learning model with gnn for eeg-emg-based stroke rehabilitation bci," *Frontiers in Neuroscience*, vol. 17, Apr 2023.

[4] Iqram Hussain and Rafsan Jany, "Interpreting stroke-impaired electromyography patterns through explainable artificial intelligence," *Sensors*, vol. 24, no. 5, pp. 1392, Feb 2024.

[5] Margarita Vergara, J.L. Sancho-Bru, V. Gracia-Ibáñez, and A. Pérez-González, "An introductory study of common grasps used by adults during performance of activities of daily living," *Journal of Hand Therapy*, vol. 27, no. 3, pp. 225–234, Jul 2014.

[6] Ian Lenz, Honglak Lee, and Ashutosh Saxena, "Deep learning for detecting robotic grasps," *International Journal of Robotics Research*, vol. 34, 01 2013.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009, pp. 248–255.

[8] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," in *CVPR*, 2024.

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," 2015.

[10] Harsh Yadav, "Computer vision: Convolution basics," Jul 2022.

[11] Charlie Goldstraw, "Convolutional autoencoders," 23AD.

[12] Suvaditya Mukherjee, "The annotated resnet-50," Aug 2022.

[13] Haoran Zhu, Boyuan Chen, and Carter Yang, "Understanding why vit trains badly on small datasets: An intuitive perspective," 2023.

[14] Ian Lenz, Honglak Lee, and Ashutosh Saxena, "Deep learning for detecting robotic grasps," *Robotics: Science and Systems IX*, Jun 2013.

[15] Ghazal Ghazaei, Federico Tombari, Nassir Navab, and Kianoush Nazarpour, "Grasp type estimation for myoelectric prostheses using point cloud feature learning," 2019.