Image Segmentation Using U-Net

Sahil Roshen (22b2542)
Afnan Abdul Gafoor (22b2505)
Ram Kandalkar (20D110018)

Project Guide: Manjesh Kumar Hanawal, IIT Bombay
DS 303 Course Project

May 2, 2025

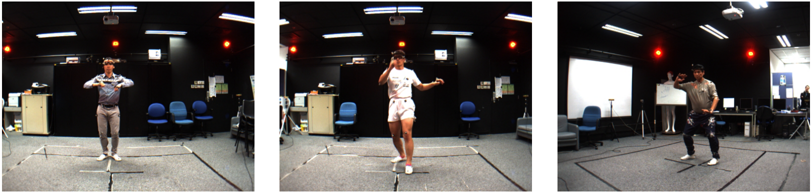
Overview

Image Segmentation Using U-Net

- **Goal:** Accurately segment humans from complex backgrounds using a deep learning model trained on real-world action images.
- **Why It Matters:**
  - Human segmentation is foundational for tasks like activity recognition, motion tracking, and augmented reality.
  - Traditional methods often fail under pose variation, occlusion, or dynamic movements.
- **Our Approach:**
  - Leverage the U-Net architecture for precise, pixel-level segmentation.
  - Train and evaluate the model on the MADS dataset — a diverse collection of studio-captured human actions.

Dataset Origin & MADS Overview

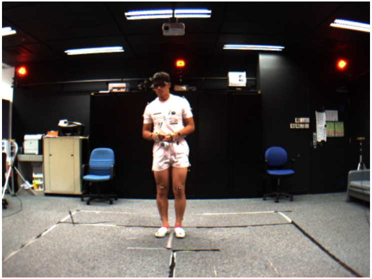
- **Dataset:** Martial Arts, Dancing and Sports (MADS)
  - Contains a variety of studio-captured human actions
  - Actions: Tai-chi, Kata, Hip-hop, Jazz, Basketball, Volleyball, Tennis, Badminton
- **Original Source:** Visual Analysis Lab, City University of Hong Kong
- **Mirror:** Kaggle repository



Random samples from the dataset

Dataset Statistics & Example

- **Sample Count:** 1,192 image–mask pairs (PNG format)
- **Resolution:** 512×384 pixels
- **Annotations:** Binary masks ($1 =$ person, $0 =$ background)



RGB input

Segmentation mask

Example image and its corresponding mask

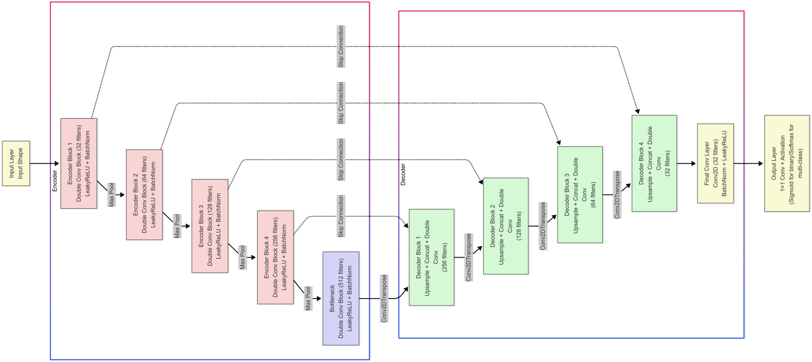
Related Work – RCNN and Pose Estimation

- **Mask R-CNN:** A popular extension of Faster R-CNN that adds a segmentation head to detect and segment objects at the instance level.
  - Achieves high accuracy in human segmentation tasks with clear boundaries.
  - Requires large annotated datasets and is computationally expensive.
- **Human Pose Estimation (e.g., OpenPose, HRNet):**
  - Focuses on detecting human joint keypoints rather than segmenting body regions.
  - Effective for tracking and activity recognition but lacks dense spatial coverage.
- **Our Approach:**
  - Uses U-Net for efficient, pixel-level human segmentation.
  - Balances accuracy and speed on a compact, action-oriented dataset (MADS).

Data Loading & Preprocessing

- **Pre-allocate** NumPy arrays for images and masks to enable fast, vectorized loading.
- **Load Images** with Keras and resize to 256×256 in RGB format for consistent network inputs.
- **Load Masks** as 256×256 grayscale images to produce uniform single-channel targets.
- **Normalize** all pixel values to [0,1] to speed up convergence and stabilize training.
- **Split Data**: first 10 samples for testing; remaining samples for training (20 % validation) due to limited data.

U-Net Architecture



Overview of the U-Net model: contracting encoder (pink), bottleneck (blue), and expansive decoder (green) with skip connections

U-Net Architecture Overview (Part 1)

- The encoder has two 3×3 convolutions at each level to extract detailed features.
- Max pooling (2×2) reduces spatial dimensions, allowing the model to focus on broader patterns.
- Batch normalization keeps activations stable, which speeds up training.
- LeakyReLU allows small gradients even for negative inputs, avoiding dead neurons.
- In the bottleneck, two 3×3 convolutions with 512 filters capture high-level global features.
- This deepest layer connects the encoder and decoder paths.

U-Net Architecture Overview (Part 2)

- The decoder upsamples features using transpose convolutions to restore image size.
- Skip connections bring encoder features directly into the decoder, recovering fine details.
- Filter counts decrease as we move up ($256 \rightarrow 128 \rightarrow 64 \rightarrow 32$), reducing complexity gradually.
- This symmetrical structure helps the model learn both context and precision.
- A final 1×1 convolution outputs one channel per class.
- A sigmoid activation gives pixel-wise probabilities for binary segmentation.

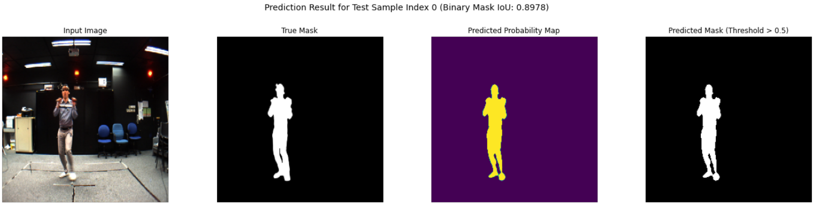
Training Configuration

- **Loss Function:** Jaccard Loss ($-$IoU), encourages accurate overlap of predicted and true masks.
- **Intersection over Union (IoU):** Measures pixel-wise overlap between predicted and ground truth masks. Computed as:

$$\text{IoU} = \frac{|\text{Prediction} \cap \text{Ground Truth}|}{|\text{Prediction} \cup \text{Ground Truth}|}$$

  Higher IoU indicates better segmentation accuracy.
- **Optimizer and Learning Rate:** Adam optimizer with a learning rate of 1×10^{-3} for efficient training.
- **Training Configuration:** Model trained for 20 epochs using a batch size of 32 with a 20% validation split.

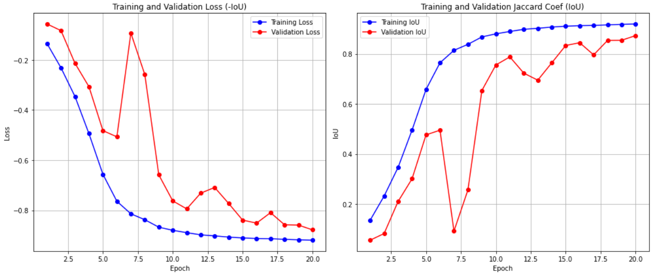
Qualitative Results



From left to right: Input image, ground truth mask, predicted mask, and overlay. The model effectively captures fine structures and body contours.

Quantitative Results & Training Curves

- **Test Set IoU (10 samples)**: **0.9022**



Train & Validation Loss and IoU

Validation IoU improves over time but shows fluctuations, indicating some variance or minor overfitting.

Conclusion

- Developed a U-Net-based model for human segmentation using the MADS dataset.
- Achieved high accuracy (IoU $= 0.9022$) despite limited training data and diverse human actions.
- Demonstrated that lightweight architectures like U-Net can be effective for action-specific segmentation tasks.
- Future work may explore:
  - Incorporating attention mechanisms for better focus on limbs and fine structures.
  - Extending the model to handle multiple subjects or real-world backgrounds.
  - Comparing performance with Mask R-CNN and pose estimation pipelines.

References

Segmentation Full Body MADS Dataset.
Kaggle Dataset Repository (link)

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross B. Girshick (2017).
Mask R-CNN.
Proceedings of the IEEE International Conference on Computer Vision (ICCV), 2017.
arXiv:1703.06870

Olaf Ronneberger, Philipp Fischer, and Thomas Brox (2015).
U-Net: Convolutional Networks for Biomedical Image Segmentation.
Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015, pp. 234-241.
arXiv:1505.04597

Zhe Cao, Tomas Simon, Shih-En Wei, Yaser Sheikh (2017).
Realtime Multi-Person 2D Pose Estimation Using Part Affinity Fields.
CVPR, 2017.
arXiv:1611.08050