Semantic Neural View Synthesis for Key Content Preservation in Horizontal-to-Vertical Video Conversion

Dipanita Chakraborty*, Minoru Okada*, and Kosin Chamnongthai†
* Nara Institute of Science and Technology, Nara 630-0192, Japan
E-mail: chakraborty.dipanita@naist.ac.jp
† King Mongkut's University of Technology Thonburi, Bangkok 10140, Thailand

***Abstract*—As short video content consumption on smartphones continues to grow, converting traditional 16:9 horizontal videos into 9:16 vertical formats has become essential. However, simple cropping or existing deep learning-based subject-aware techniques often lead to the loss of important visual content, especially when key objects are positioned near the frame edges. This research proposes a deep generative learning-based semantic novel view synthesis framework to preserve essential content during horizontal-to-vertical (H2V) video conversion. The system leverages an instance segmentation network to classify subjects and objects in both the foreground and background of video scenes. An autoencoder-based action recognition network then extracts semantic information to identify subject actions and their spatiotemporal correlations with prioritized key objects in each frame. A neural view synthesis module, based on a generative adversarial network (GAN), reconstructs the background along with the key objects and reframes the scene to fit the vertical aspect ratio. Additionally, a post-processing step refines any missing regions caused by reframing to achieve the optimal key content preservation. Experiments on the benchmark video dataset show that the proposed method outperforms traditional scaling and center-cropping techniques in preserving object spatiotemporal relationships and maintaining visual coherence.**

I. Introduction

The widespread use of smartphones has led to a surge in the consumption of short-form videos in vertical (9:16) formats [**1**]. However, a significant portion of legacy and cinematic video content remains in the traditional horizontal (16:9) format. Adapting such content to vertical orientation while maintaining semantic integrity poses unique challenges. Naive approaches such as uniform scaling, center cropping, or even basic subject-aware cropping frequently lead to the exclusion of key subjects or scene elements—especially when objects of interest are located near the horizontal frame boundaries.

To address these limitations, this work proposes a semantic novel view synthesis framework based on deep generative learning. The objective is to intelligently convert horizontal-to-vertical (H2V) videos by understanding both the semantics and spatiotemporal dynamics of a scene. The proposed framework is a multi-stage pipeline that first segments all instances in an image, then specifically identifies human-object pairs engaged in meaningful interactions, reconstructs the background where these interactions occur, and finally re-frames the involved objects for specific display formats. This comprehensive approach moves beyond simple detection, enabling a deeper contextual understanding and offering tools for novel image synthesis and adaptation. The key contributions of this work include:

1) An Instance Segmentation to detect and separate foreground elements (subjects) from the background.
2) A Variational Autoencoder (VAE) for learning and detecting human-object relationships by extracting spatiotemporal semantics and prioritizing key object interactions to bridging the gap between low-level visual features and high-level semantic interactions.
3) A novel GAN-based synthesis module for conditional scene generation, capable of background reconstruction, and re-integrating foreground objects.
4) A refinement post-processor for re-framing contextually relevant objects from horizontal (16:9) into the desired vertical aspect ratio (9:16), demonstrating practical application in media generation.

In this article, Section III describes the proposed methodology in detail and Section IV explains implementation and experimental evaluation. Finally, Section V concludes the paper and outlines future work.

II. Related Work

A. Instance Segmentation

Instance segmentation, the task of detecting and segmenting each object instance in an image, has seen significant advancements. Early methods often relied on two-stage approaches like Mask R-CNN [**2**]. More recently, single-stage detectors, such as the YOLO family, have been extended to perform instance segmentation. YOLOv8-seg [**3**], a state-of-the-art model, offers a compelling balance of speed and accuracy, making it suitable as the foundational perception module in our pipeline.

B. Human-Object Interaction (HOI) Detection

HOI detection aims to identify triples of (human, verb, object) in an image. Early works utilized handcrafted features and graphical models [**4**]. With the advent of deep learning, methods have evolved to employ CNN features [**5**], often coupled with attention mechanisms or graph neural networks

979-8-3315-7206-8/25/$31.00 ©2025 IEEE

to model the interactions. VAEs, as probabilistic generative models, offer a unique perspective for learning latent representations of complex data, which can be adapted for relationship detection by modeling the distribution of human-object pairs and their actions.

C. Reconstruction and Novel View Synthesis

Image inpainting, the task of filling in missing or corrupted parts of an image, has been revolutionized by GANs [**6**]. Pix2Pix [**7**], a conditional GAN, has proven particularly effective for image-to-image translation tasks, including inpainting, by learning a mapping from an input image to a corresponding output image. The adversarial loss, combined with a pixel-wise reconstruction loss, enables the generation of semantically consistent and visually plausible content. This capability is critical for our framework to reconstruct backgrounds. While classical novel view synthesis often implies changes in camera viewpoint, our approach extends this concept to *semantic view synthesis*, where the "new view" is a re-composition and re-framing of a scene based on the intelligent arrangement of foreground objects within a generated background, adapted for a specific aspect ratio. This is distinct from simple image inpainting, as it involves a creative re-composition of scene elements.

In summary, our research integrates and builds upon these advanced techniques, creating an effective pipeline for a deeper understanding of visual content.

III. Methodology

Our framework consists of five distinct, interconnected stages, each leveraging specialized deep learning models. The overall pipeline is depicted in Fig. 1.

A. Stage 1: Instance Segmentation with YOLOv8-seg

The initial step involves accurately segmenting all instances in the video frame. We employ the state-of-the-art YOLOv8-seg model [**3**], pre-trained on the COCO dataset, and fine-tune it on our custom COCO-formatted dataset. The YOLOv8-seg network architecture is shown in Table I.

Training: The training of YOLOv8-seg follows standard procedures as implemented in the machine learning library.

$$\mathcal{L}_{YOLO} = \mathcal{L}_{cls} + \mathcal{L}_{reg} + \mathcal{L}_{seg} \tag{1}$$

where $\mathcal{L}_{cls}$ is the classification loss, $\mathcal{L}_{reg}$ is the bounding box regression loss, and $\mathcal{L}_{seg}$ is the segmentation mask loss.

Inference and Output: After training, we perform inference on all images (train, validation, and test sets). For each image, the model outputs a set of detected instances, each comprising a bounding box ($[x_1, y_1, x_2, y_2]$), a precise polygonal segmentation mask, a confidence score, and a class ID. These segments form the foundation for subsequent stages.

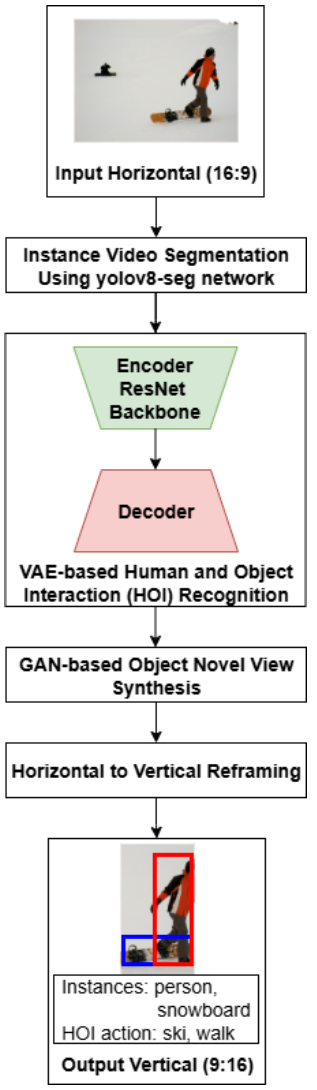

Fig. 1. Proposed semantic neural view synthesis for key content preservation in Horizontal-to-Vertical video conversion.

TABLE I
YOLOv8-seg Architecture Overview

Stage	Layer Type	Output Shape	Details
Backbone	Conv	$B \times 32 \times 160 \times 160$	3x3, stride 2
	C2f	$B \times 64 \times 80 \times 80$	1 repeat
	C2f	$B \times 128 \times 40 \times 40$	2 repeats
	C2f	$B \times 256 \times 20 \times 20$	2 repeats
	C2f	$B \times 512 \times 10 \times 10$	1 repeat
Neck (FPN+PAN)	Upsample + Concat	$B \times 256 \times 20 \times 20$	with C2f(256)
	C2f	$B \times 256 \times 20 \times 20$	1 repeat
	Upsample + Concat	$B \times 128 \times 40 \times 40$	
	C2f	$B \times 128 \times 40 \times 40$	1 repeat
Detect Head	Detect	$B \times N_a \times$ (4 + C + 32)	box + cls + mask embedd
	NMS	-	Non-Max Suppress
Segment Head	Proto	$B \times 32 \times 160 \times 160$	mask generate
	Mask	$B \times C \times 160 \times 160$	sigmoid (mask × embed)

Hyperparams: Input = 224×224, Batch size = 16, epochs = 50, Learning rate = 0.01, Optimizer = SGD with momentum 0.937, Weight decay = 5e-4, Loss comp.: box, class, mask. class no. = 80.

B. Stage 2: Human-Object Relationship Detection with VAE

This stage focuses on identifying semantic relationships between humans and objects using a Variational Autoencoder (VAE). The VAE is trained on the VCOCO dataset, which provides annotations for human-object pairs and their associated actions.

1) VCOCO Label Preprocessing: We parse the VCOCO annotations to extract tuples of $(image\_id, human\_bbox, object\_bbox, action\_label)$. A critical step is mapping the VCOCO object classes to the class IDs output by YOLOv8-seg, ensuring consistency across stages. This involves creating a lookup table for object names to YOLOv8-seg class IDs.

2) VAE Model Design: The VAE is designed to learn a compressed, meaningful latent representation of human-object interactions. Table II demonstrates the VAE model architecture.

Input Representation: For each human-object pair, we construct an input vector by concatenating visual features extracted from the cropped human and object bounding box regions with spatial features. A pre-trained Convolutional Neural Network (CNN) (ResNet-18 as a feature extractor) processes the resized cropped regions.

$$\mathbf{f}_{human} = \text{CNN}(\text{Crop}(\text{Image}, \text{human\_bbox})) \tag{2}$$

$$\mathbf{f}_{object} = \text{CNN}(\text{Crop}(\text{Image}, \text{object\_bbox})) \tag{3}$$

Spatial features $\mathbf{f}_{spatial}$ encode relative positions, sizes, and potentially Intersection over Union (IoU) of the bounding boxes. The final VAE input is $\mathbf{x} = [\mathbf{f}_{human}, \mathbf{f}_{object}, \mathbf{f}_{spatial}]$.

Loss Function: The Variational Autoencoder (VAE) is optimized using a composite loss function:

$$\mathcal{L}_{VAE} = \mathcal{L}_{reconstruction} + \beta \cdot \mathcal{L}_{KL} + \mathcal{L}_{classification} \tag{4}$$

where:

- **Reconstruction Loss** encourages the decoder to faithfully reconstruct the input $\mathbf{x}$ from the latent encoding:

$$\mathcal{L}_{reconstruction} = ||\mathbf{x} - \text{Decoder}(\text{Encoder}(\mathbf{x}))||_2^2 \tag{5}$$

  This is the mean squared error (MSE) between the original input and its reconstruction.
- **KL (Kullback–Leibler) Divergence Loss** regularizes the learned latent distribution $q(\mathbf{z}|\mathbf{x})$ to be close to a standard normal prior $p(\mathbf{z}) = \mathcal{N}(0, I)$:

$$\mathcal{L}_{KL} = -0.5 \sum_{i=1}^{D} \left(1 + \log \sigma_i^2 - \mu_i^2 - \sigma_i^2\right) \tag{6}$$

  where μ_i and σ_i are the mean and standard deviation of the latent variable $\mathbf{z}$.
- **Classification Loss** ensures that the latent representation is informative for downstream classification tasks:

$$\mathcal{L}_{classification} = -\sum_{k} y_k \log(\hat{y}_k) \tag{7}$$

  which is the categorical cross-entropy between the true labels y_k and predicted probabilities $\hat{y}_k$.

β is a hyperparameter balancing reconstruction fidelity and latent space regularization, and D is the dimensionality of the latent space.

TABLE II
VARIATIONAL AUTOENCODER (VAE) ARCHITECTURE OVERVIEW

Stage	Layer	Shape	Details
Encoder	FC1	$[B,$ *input_dim*$]$	FC + ReLU
	FC_	$[B,$ *hid._dim*$]$	Mean vector
	FC_logvar	$[B,$ *hid._dim*$]$	Log vector
	Reparam.	$\mu, \log \rightarrow z$	$z = \mu + \epsilon \cdot e^{0.5 \log}$
Decoder	FC_decode1	$[B,$ *latent_dim*$]$	FC + ReLU
	FC_out	$[B,$ *hid._dim*$]$	Reconst. vector
Classifier	FC_cls	$[B,$ *lat._dim*$]$	Action (CrossEntropy)
Loss	–	–	$\mathcal{L} = \text{MSE} + \text{KLD} + \gamma\,\text{CE}$

Hyperparams: input = 224×224, Optimizer = Adam, action = 46 HOI classes, batch size = 32, learning rate = 1×10^{-4}, epochs = 50.

C. Stage 3: Filtering and Saving Relevant Segmentations

This stage integrates the outputs of Stage 1 (YOLOv8 segmentations) and Stage 2 (VAE-detected relationships) to extract only the human and object instances that are part of a recognized interaction.

Matching Strategy: For each image, we iterate through the relationships detected by the VAE. For each human-object pair from the VAE, we search for a corresponding human and object segmentation from YOLOv8-seg. Matching is performed using Intersection over Union (IoU) between bounding boxes, with a threshold (e.g., IoU $>$ 0.5), and validated by class ID consistency (e.g., YOLO detection is 'person' for human, and matches expected object class for the object).

Output: The filtered segmentation masks and their bounding box information are saved for each image, forming a curated dataset of meaningful human-object interactions.

D. Stage 4: GAN for Missing Region Reconstruction

This stage aims to semantically reconstruct the background of an image after removing the detected human-object interaction. This creates an "empty" context that can be reused or analyzed. The GAN architecture is summarized in Table III.

1) GAN Dataset Preparation: From the filtered segmentations (Stage 3), we create paired training data: noitemsep,topsep=0pt

- **GAN Input (Masked Image):** The original image with the pixels corresponding to the human and related object segmentation masks set to black (or random noise).
- **GAN Target (Original Image):** The untouched original image.

2) GAN Architecture and Training: We adopt a Pix2Pix-like Conditional Generative Adversarial Network (cGAN) [7] for this image-to-image translation task.

TABLE III
GAN ARCHITECTURE

Component	Layer	Output
Generator (U-Net)		
Input	Masked RGB Image	$3 \times 256 \times 256$
Initial Down	Conv2D (64, 4×4, s=2), LeakyReLU	$64 \times 128 \times 128$
Down 1–6	$6 \times$ [Conv2D, BatchNorm, LeakyReLU]	$512 \times 2 \times 2$
Bottleneck	Conv2D (512, 4×4, s=2), ReLU	$512 \times 1 \times 1$
Up 1–6	$6 \times$ [ConvTranspose2D, BatchNorm, ReLU + skip connection]	$64 \times 128 \times 128$
Final Up	ConvTranspose2D (3, 4×4, s=2), Tanh	$3 \times 256 \times 256$
Discriminator (PatchGAN)		
Input	Concatenated masked + target/generated image	$6 \times 256 \times 256$
Conv 1	Conv2D (64, 4×4, s=2), LeakyReLU	$64 \times 128 \times 128$
Conv 2	Conv2D (128, 4×4, s=2), BatchNorm, LeakyReLU	$128 \times 64 \times 64$
Conv 3	Conv2D (256, 4×4, s=2), BatchNorm, LeakyReLU	$256 \times 32 \times 32$
Conv 4	Conv2D (512, 4×4, s=1), BatchNorm, LeakyReLU	$512 \times 31 \times 31$
Output	Conv2D (1, 4×4, s=1)	$1 \times 30 \times 30$

Hyperparams: Learning rate = 0.0002; Optimizer = Adam; β_1 = 0.5; β_2 = 0.999; Batch size = 1; Image = 256×256; epochs = 200; L1 loss weight = 100; Batch Normalization; Activations = LeakyReLU , ReLU (decoder), Tanh (Generator output).

- **Generator (G):** A U-Net architecture that takes the masked image as input and generates a reconstructed image.
- **Discriminator (D):** A PatchGAN discriminator that classifies pairs of (masked image, real image) or (masked image, generated image) as real or fake.

Loss Functions: The GAN is trained with an adversarial loss and an L1 reconstruction loss.

$$\begin{aligned}\mathcal{L}_{GAN}(G,D) = E_{\mathbf{x},\mathbf{y}\sim p_{data}(\mathbf{x},\mathbf{y})}[\log D(\mathbf{x},\mathbf{y})] \\ + E_{\mathbf{x}\sim p_{data}(\mathbf{x})}[\log(1 - D(\mathbf{x}, G(\mathbf{x})))]\end{aligned} \tag{8}$$

$$\mathcal{L}_{L1}(G) = E_{\mathbf{x},\mathbf{y}\sim p_{data}(\mathbf{x},\mathbf{y})}[||\mathbf{y} - G(\mathbf{x})||_1] \tag{9}$$

The total generator loss is:

$$\mathcal{L}_G = \mathcal{L}_{GAN}(G,D) + \lambda \mathcal{L}_{L1}(G) \tag{10}$$

where λ balances the adversarial and reconstruction terms.

E. Stage 5: Object Re-framing (Horizontal to Vertical)

The final stage involves isolating the relevant human and object instances and re-framing them into a vertical aspect ratio.

1) **Cropping and Masking:** The human and object instances are cropped from the original image using their bounding boxes, and their segmentation masks are applied to isolate the object from its original background.
2) **Aspect Ratio Calculation:** A target vertical aspect ratio (e.g., 9:16) is defined. The dimensions of a new canvas are calculated to fully contain the cropped object while maintaining the target aspect ratio.
3) **Padding:** The isolated object is pasted onto this new, vertically oriented canvas, typically centered, with transparent padding applied to fill the remaining space. This ensures the object is fully visible within the desired vertical frame without distortion.

We discussed the implementation of the proposed methodology and evaluated end-to-end performance in the section IV.

IV. EXPERIMENTAL RESULTS

A. Dataset and Implementation Details

- **COCO Dataset:** This data is for initial YOLOv8-seg pretraining and fine-tuning [**8**]. We selected 10346 images split into training, validation, and test sets (80:10:10) and 80 labeled classes of objects.
- **VCOCO Dataset:** This data is used for training the VAE for human-object relationship detection [**9**]. We selected the same 10346 images as in the COCO dataset and split them into training, validation, and test sets (80:10:10) with 46 labeled actions between humans and objects.
- **H2V-142K:** This data is used for testing of the proposed model [**10**].
- **Hardware/Software:** The system is implemented in Python with PyTorch on an NVIDIA RTX A4000 GPU, Intel Xeon W5-2465X CPU, 64-bit OS, and 1 TB SSD.

B. Stage 1: Instance Segmentation Performance

We fine-tuned the YOLOv8-seg model on our custom dataset for 50 epochs across 80 COCO object classes. The model exhibits strong segmentation performance, as shown in Fig. 2 and the confusion matrix in Fig. 3. Performance is evaluated using Recall (R) and mean Average Precision (mAP), which reflect detection accuracy across IoU thresholds. Higher values indicate better localization and segmentation. A summary of class-wise results is presented in Table IV.

TABLE IV
INSTANCE SEGMENTATION PERFORMANCE (BOX LOSS)

Class	Recall (R) ↑	mAP (0.5:0.95) ↑
Toilet	0.995	0.796
Bed	0.722	0.658
Surfboard	0.740	0.506
Tennis Racket	0.714	0.489
Person	0.700	0.595
Motorcycle	0.737	0.535
Horse	0.750	0.588

Among the evaluated classes, **toilet** achieved the highest performance (R = 0.995, mAP = 0.796), indicating highly accurate detection. **Bed** also performed well (R = 0.722, mAP = 0.658). **Surfboard** and **motorcycle** showed moderate accuracy, while **tennis racket** had the lowest scores, likely due to visual ambiguity or limited training data. **Person** and **horse** classes demonstrated balanced results with stable segmentation quality. The Shadows of persons were falsely detected as 'person' instances. This limitation can be addressed by incorporating shadow detection techniques, which will be considered in future work.

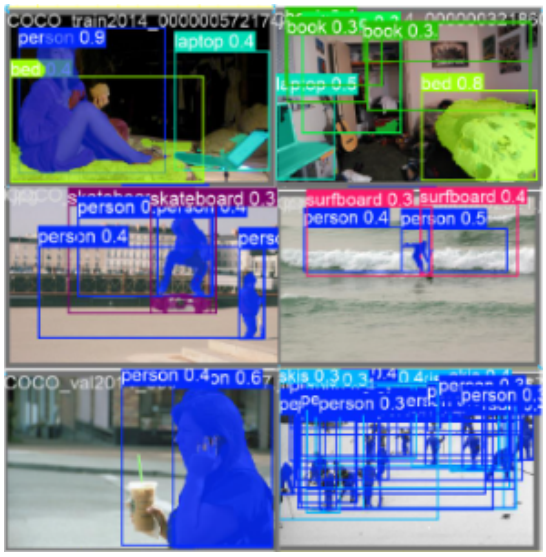

Fig. 2. Qualitative examples of instance segmentation results from the YOLOv8-seg network.

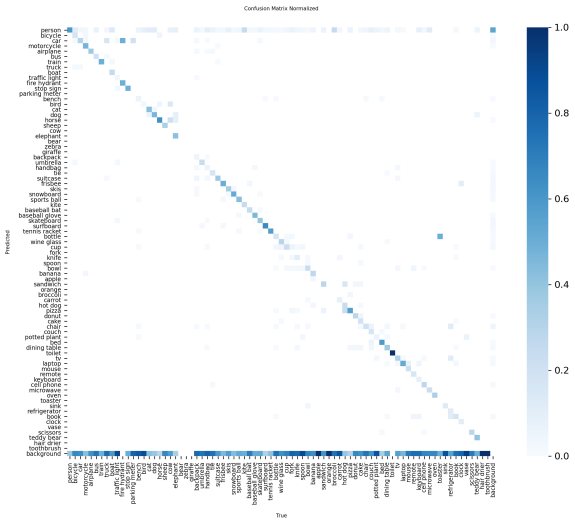

Fig. 3. Confusion matrix of instance segmentation results of 80 classes using the YOLOv8-seg network.

C. Stage 2: Human-Object Relationship Detection Performance

The VAE was trained for 50 epochs. We evaluate the classification accuracy of the VAE's classifier head for relationship detection.

TABLE V
VAE-BASED HOI DETECTION PERFORMANCE (LOSS VALUES)

Loss Type	Value
$\mathcal{L}_{reconstruction}$	0.074
$\mathcal{L}_{classification}$	0.141
Total Loss	**0.215**

Table V presents the final training loss values for the Variational Autoencoder after 50 epochs. The reconstruction loss ($\mathcal{L}_{reconstruction}$) of 0.074 indicates a low error in reconstructing the input features from the latent space, suggesting that the VAE effectively learned a compact representation of human-object interactions. The classification loss ($\mathcal{L}_{classification}$) of 0.141 reflects the VAE's ability to classify the action associated with the human-object pairs. VAE outputs are matched to YOLO detections with the highest IoU and consistent category. When VAE outputs overlaps with the same YOLO detection, the match with the highest YOLO confidence is retained while others are discarded and unmatched VAE outputs are treated as false positives.

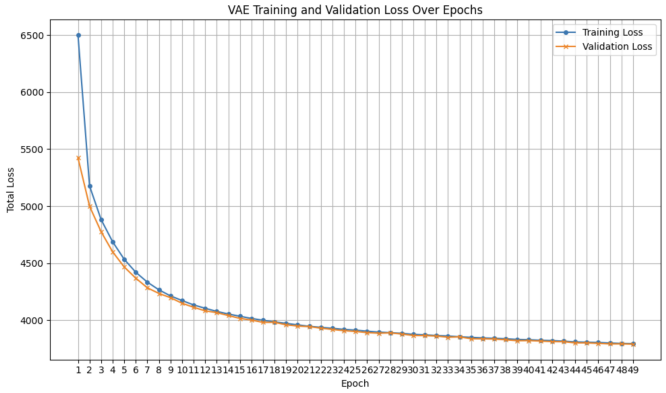

Fig. 4. Training vs validation loss curve using the VAE model.

D. Stage 4: Missing Region Reconstruction Performance

In the final stage, a GAN-based inpainting model is employed to reconstruct the missing vertical regions, producing visually plausible content that aligns semantically and texturally with the surrounding image. The model leverages contextual cues to hallucinate realistic structures and fine-grained textures in occluded areas. Qualitative results in Fig. 5 highlight the effectiveness of the proposed GAN framework. The generated outputs demonstrate strong consistency in both texture and geometry, preserving visual coherence across the synthesized vertical content. Errors from earlier stages can result in weak visual cues in complex or low-intensity scenarios which will be part of the future work.

E. End-to-End Pipeline Evaluation & Qualitative Results

Finally, Fig. 6 presents qualitative results demonstrating the full pipeline's capability, from initial segmentation to relationship detection, removal, reconstruction, and final reframing. Table VI demonstrates the performance comparison between the H2V framework [**10**] and the proposed method.

TABLE VI
COMPARISON WITH THE EXISTING H2V FRAMEWORK

Method	Recall (%)
H2V [**10**]	69.53
Proposed	74.00

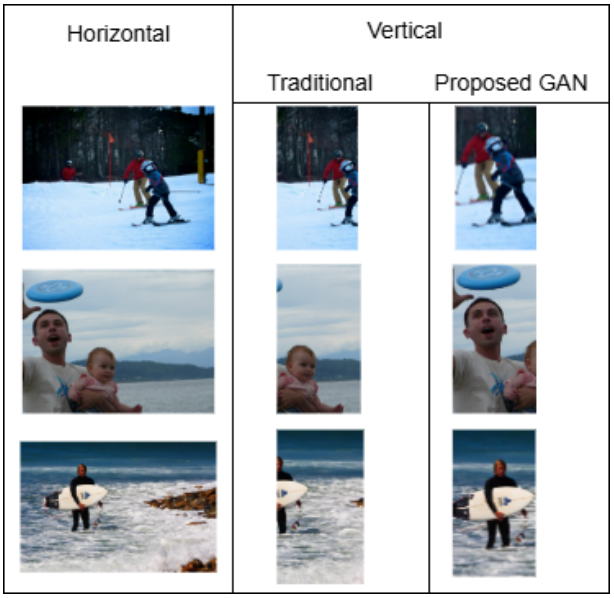

Fig. 5. Qualitative analysis of GAN model outputs

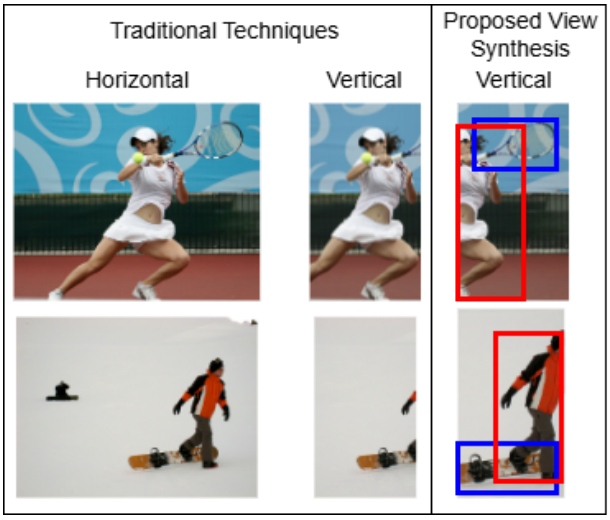

Fig. 6. End-to-end results of Human and object relation-aware (HOI action) view synthesis for Horizontal to Vertical Frame Conversion.

Fig. 5 visually demonstrates our framework's cohesive operation, showcasing the complete transformation from input to HOI action identification, novel view synthesis, and final vertical re-framing. This confirms the system's ability to semantically understand and adapt scenes. Quantitatively, Table VI shows the proposed method's efficiency with a 74.00% Recall, notably surpassing the "H2V framework" [**10**] at 69.53%. Some limitations were observed. HOI detection struggles with ambiguity when a single person performs multiple actions; future work will incorporate higher-level semantic features. Furthermore, while some instance segmentation exhibited low IoU, causing pixel loss, our GAN-based novel view synthesis module effectively reconstructs these regions, mitigating visual impact on the final scene.

V. Conclusion

This paper introduced a novel multi-stage deep learning framework for the comprehensive analysis and novel view synthesis of human-object relationships in images. By combining YOLOv8-seg for precise instance segmentation, a VAE for semantic relationship detection, and a Pix2Pix GAN for high-fidelity image inpainting, we demonstrated an end-to-end pipeline capable of understanding complex visual interactions and adapting visual content based on this understanding. The ability to accurately detect relationships, reconstruct contextual backgrounds, and intelligently re-integrate foreground elements into new compositions for specific aspect ratios opens up new avenues for applications in dynamic content creation, virtual environments, and intelligent image editing tools. Future work will focus on extending the VAE to handle more complex, multi-object relationships, exploring alternative GAN architectures for even higher reconstruction quality, and investigating the integration of 3D information to enable more sophisticated spatial manipulations and re-contextualization of objects. Furthermore, we plan to develop a user interface to showcase the interactive capabilities of this framework.

Acknowledgment

The authors highly appreciate the anonymous reviewers from APSIPA ASC 2025 for their valuable reviews and helpful comments. This research was supported by the Network Systems Laboratory at Nara Institute of Science and Technology.

References

[1] L. Mulier, H. Slabbinck, and I. Vermeir, "This Way Up: The Effectiveness of Mobile Vertical Video Marketing, Journal of Interactive Marketing," *Journal of Interactive Marketing*, vol. 55, pp. 1-15, 2021, doi:10.1016/j.intmar.2020.12.002.

[2] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," *in IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 2, pp. 386-397, Feb. 2020, doi: 10.1109/TPAMI.2018.2844175.

[3] G. Jocher, A.Chaurasia, and J. Qiu "YOLOv8," 2023, Online[https://github.com/ultralytics/ultralytics].

[4] A. Sadhu, T. Gupta, M. Yatskar, R. Nevatia, and A. Kembhavi, "Visual Semantic Role Labeling for Video Understanding," *in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Nashville, TN, USA, 2021, pp. 5585-5596, doi: 10.1109/CVPR46437.2021.00554.

[5] G. Gkioxari, R. Girshick, P. Dollár, and K. He, "Detecting and Recognizing Human-Object Interactions," *in Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, 2018, pp. 8359-8367, doi: 10.1109/CVPR.2018.00872.

[6] I. J. Goodfellow, J. P.-Abadie, M. Mirza, B. Xu, D. W.-Farley, S. Ozair, A. Courville, Y. Bengio, " 2014, *arXiv:1406.2661v1* .

[7] P. Isola, J. -Y. Zhu, T. Zhou and A. A. Efros, "Image-to-Image Translation with Conditional Adversarial Networks," *in Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, 2017, pp. 5967-5976, doi: 10.1109/CVPR.2017.632.

[8] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dollár, Microsoft COCO: Common Objects in Context," *European Conference on Computer Vision (ECCV)*, pp. 740–755, 2014, doi:10.1007/978-3-319-10602-1_48

[9] S. Gupta and J. Malik, "Visual Semantic Role Labeling," *in Proc. Chinese Control Conference*, 2015, *arXiv:1505.04474*.

[10] T. Zhu, D. Zhang, Y. Hu, T. Wang, X. Jiang, J. Zhu, and J. Li, "Horizontal-to-Vertical Video Conversion," *IEEE Transactions on Multimedia*, vol. 24, pp. 3036-3048, 2022, doi: 10.1109/TMM.2021.3092202.