# Inpainting of Depth Images using Deep Neural Networks for Real-Time Applications

**Roland Fischer**, Janis Roßkamp, Thomas Hudcovic, Anton Schlegel, Gabriel Zachmann

University of Bremen, Bremen, Germany

r.fischer@uni-bremen.de

*ISVC 2023*

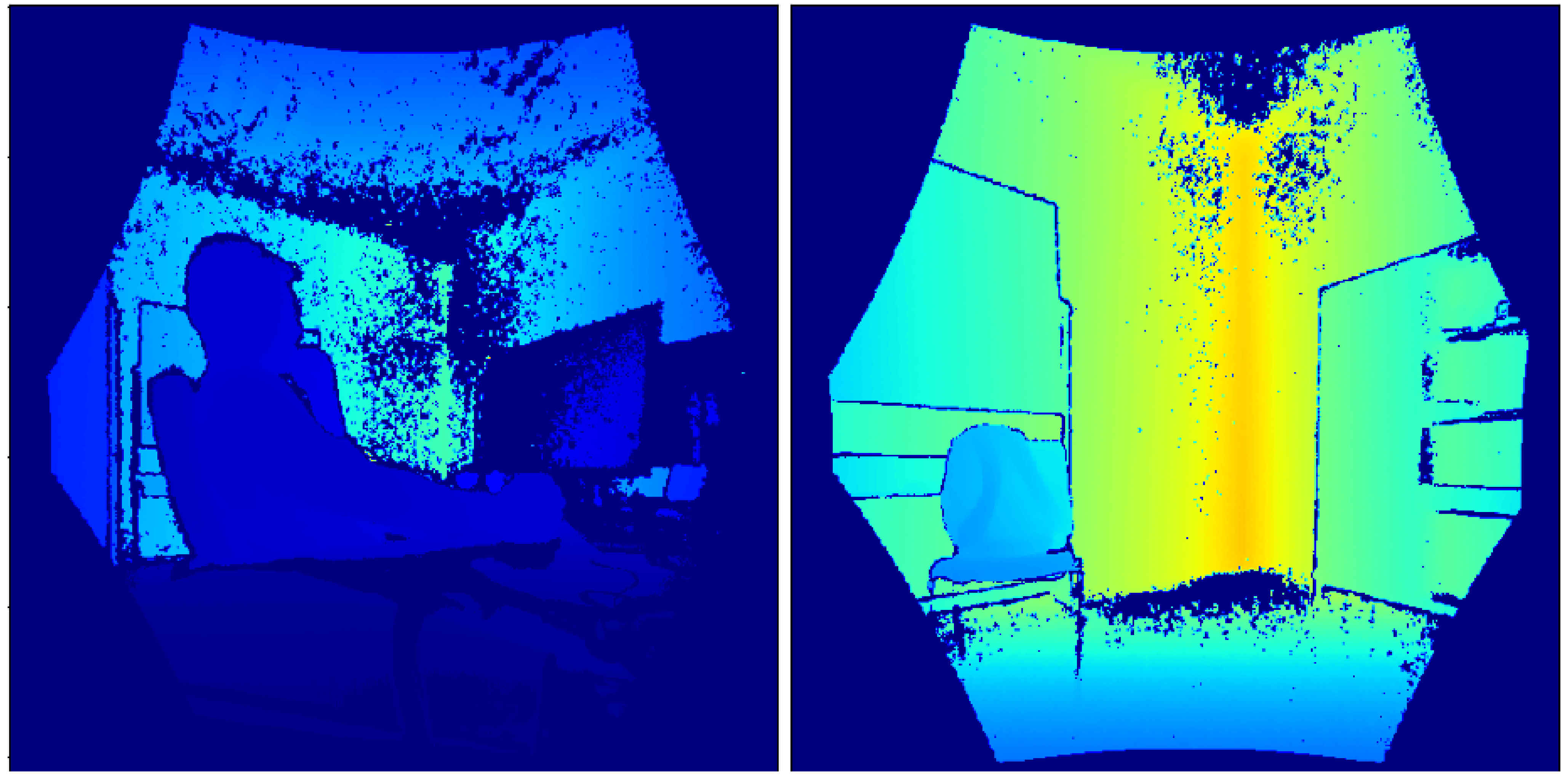
*16-18 October,* Lake Tahoe, USA

# Motivation

- RGB-D cameras/lidar widely employed
  - SLAM, object-detection, real-time avatars

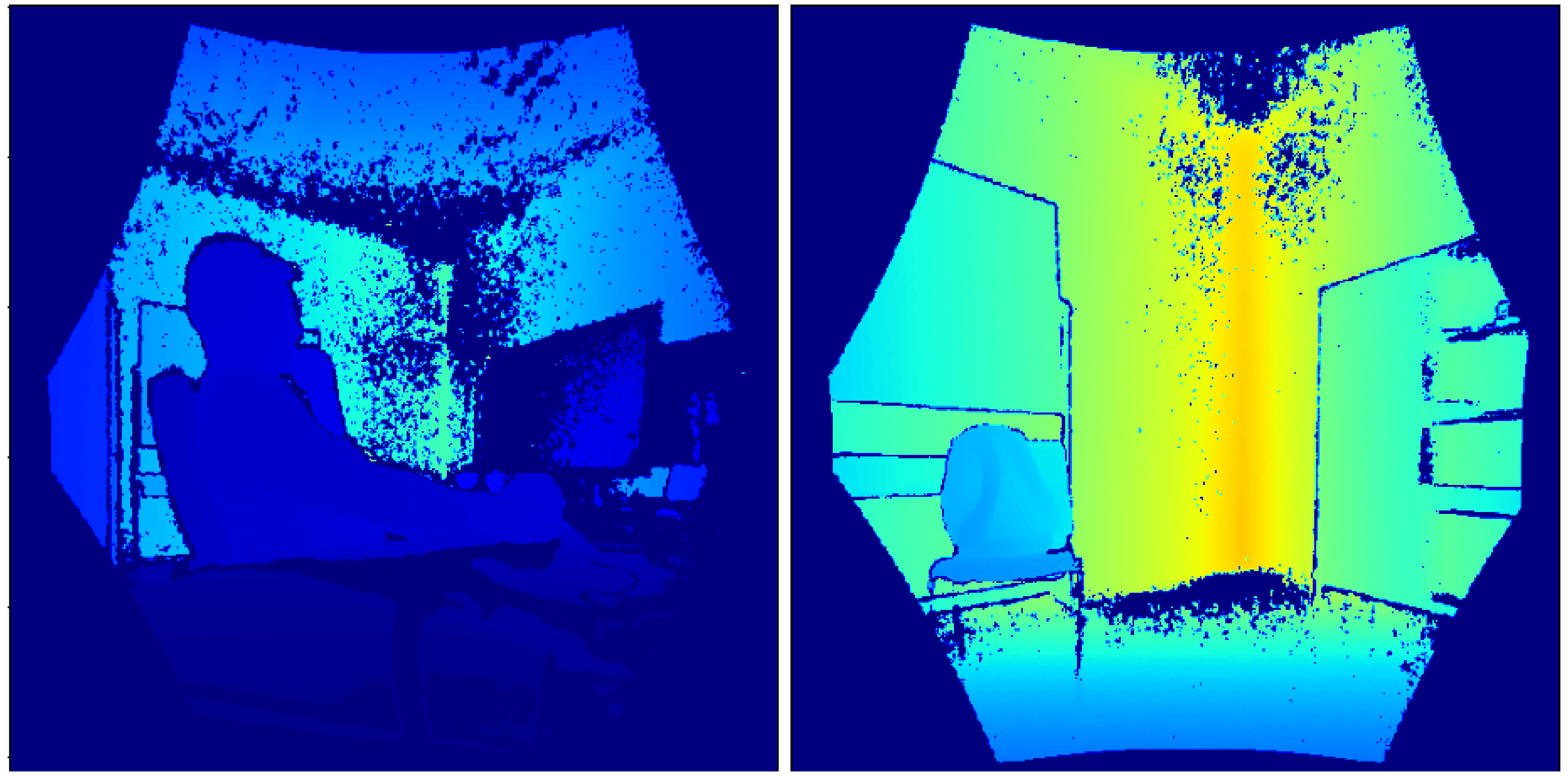
# Motivation

- RGB-D cameras/lidar widely employed
  - SLAM, object-detection, real-time avatars
- Issue: Sensor noise, holes in depth data



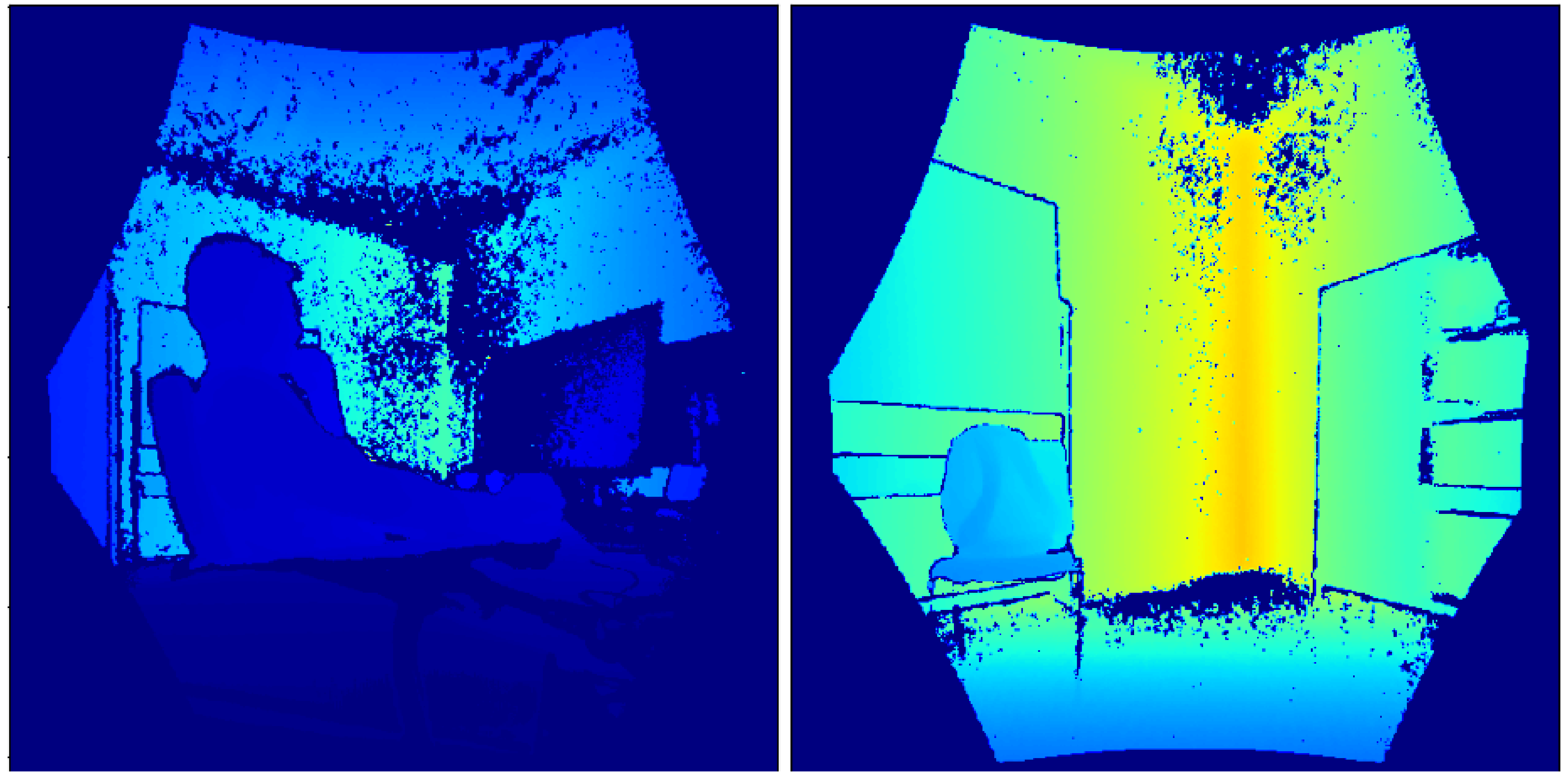
# Motivation

- RGB-D cameras/lidar widely employed
  - SLAM, object-detection, real-time avatars
- Issue: Sensor noise, holes in depth data
- Important task to reconstruct missing areas



# Motivation

- RGB-D cameras/lidar widely employed
  - SLAM, object-detection, real-time avatars
- Issue: Sensor noise, holes in depth data
- Important task to reconstruct missing areas
- High quality real-time inpainting challenging



# Related Work

- Impressive results with deep learning for various computer vision tasks

# Related Work

- Impressive results with deep learning for various computer vision tasks
- Deep learning-based inpainting mostly on color
  - Non-standard convolutions [Yu19,Ning19]
  - GANs [Isola17,Shao20]

# Related Work

- Impressive results with deep learning for various computer vision tasks
- Deep learning-based inpainting mostly on color
  - Non-standard convolutions [Yu19,Ning19]
  - GANs [Isola17,Shao20]
- Depth image inpainting
  - Still uses color guidance [Tao22,Lee22]
  - Only small holes [Jin20]

# Related Work

- Impressive results with deep learning for various computer vision tasks
- Deep learning-based inpainting mostly on color
  - Non-standard convolutions [Yu19,Ning19]
  - GANs [Isola17,Shao20]
- Depth image inpainting
  - Still uses color guidance [Tao22,Lee22]
  - Only small holes [Jin20]
- Transformer/Diffusion models very slow [Deng22,Rombach22]

# Our Contributions

- Real time depth image inpainting using deep learning
  - Without color guidance, also larger holes

# Our Contributions

- Real time depth image inpainting using deep learning
  - Without color guidance, also larger holes
- Investigated performance of various models
  - Partial convolutional U-Net
  - Patch-based GAN
  - Standard U-Net
  - LaMa

# Our Contributions

- Real time depth image inpainting using deep learning
  - Without color guidance, also larger holes
- Investigated performance of various models
  - Partial convolutional U-Net
  - Patch-based GAN
  - Standard U-Net
  - LaMa
- Detailed quantitative and qualitative evaluation
  - Two public standard datasets + self-recorded one

# Datasets

- Training:
  - NYU Depth V2 (indoor, Kinect v1)

NYUV2, color/depth [Silberman12]

# Datasets

- Training:
  - NYU Depth V2 (indoor, Kinect v1)
  - Added own synthetic holes

NYUV2, color/depth [Silberman12]

# Datasets

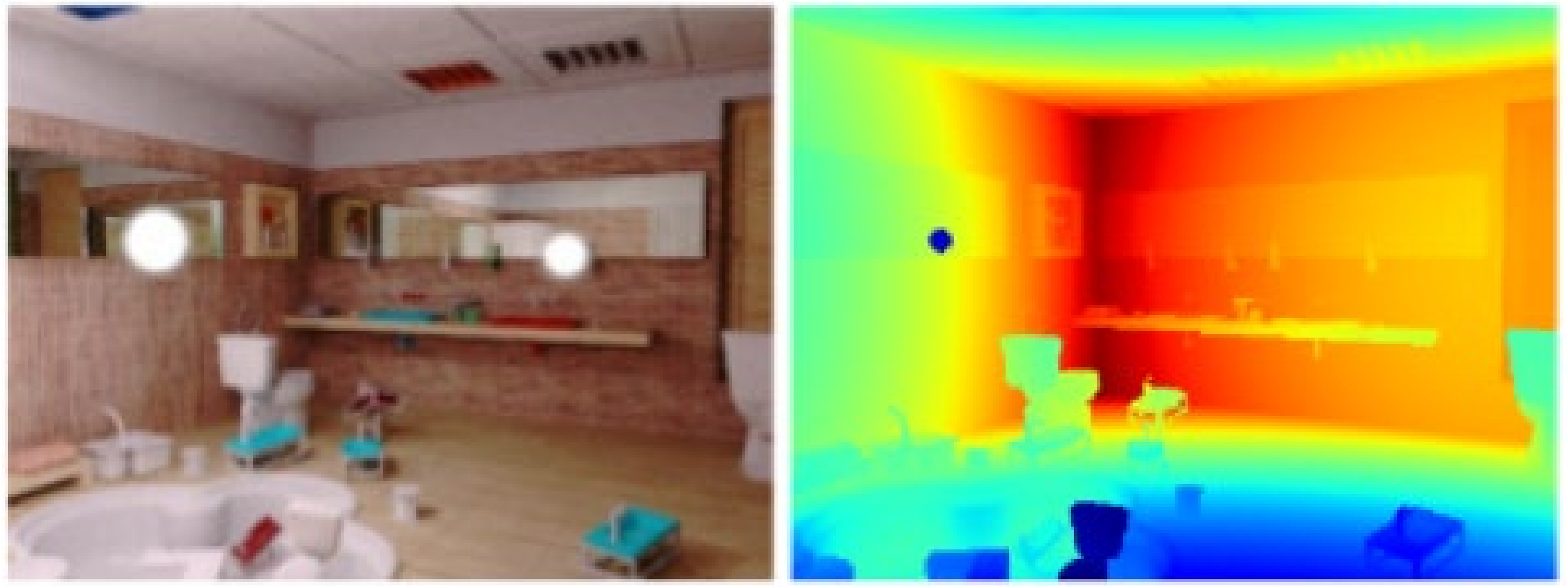
- Training:
  - NYU Depth V2 (indoor, Kinect v1)
  - Added own synthetic holes
- Evaluation
  - NYUV2

NYUV2, color/depth [Silberman12]

# Datasets

- Training:
  - NYU Depth V2 (indoor, Kinect v1)
  - Added own synthetic holes
- Evaluation
  - NYUV2
  - SceneNet RGB-D (synthetic indoor scenes, low resolution, using depth only, added holes)

NYUV2, color/depth [Silberman12]

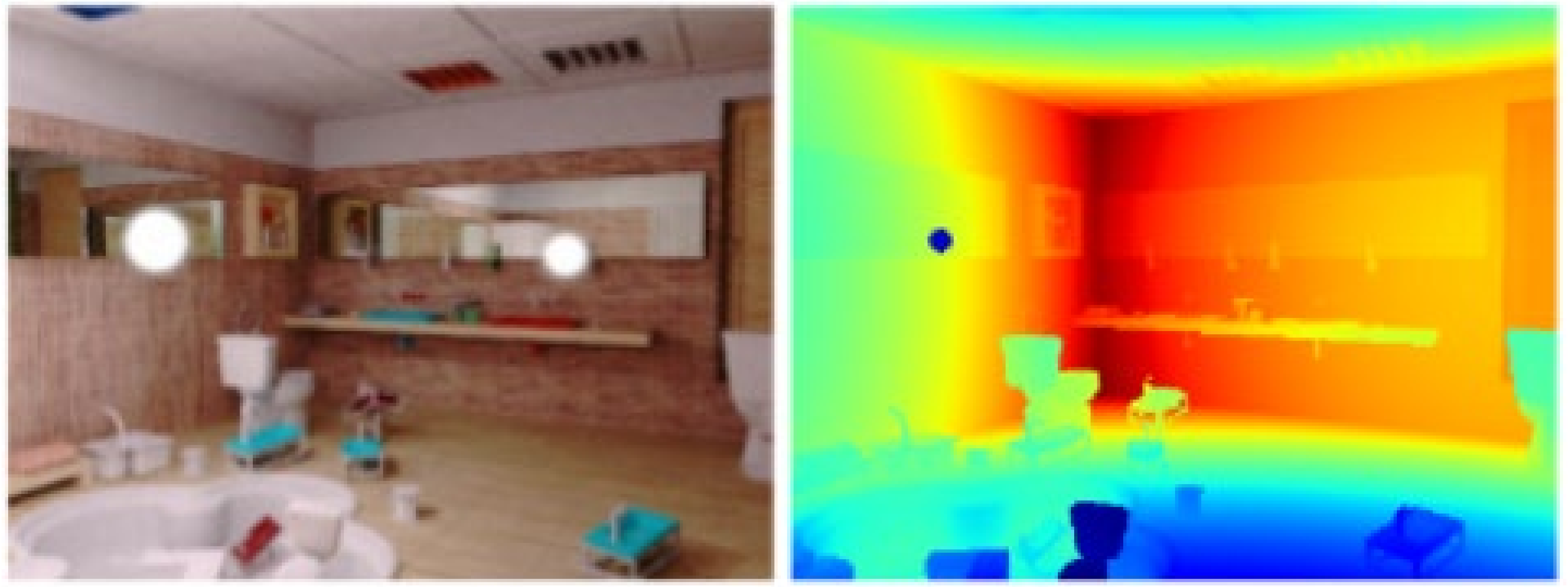


SceneNet, color/depth [McCormac16]

# Datasets

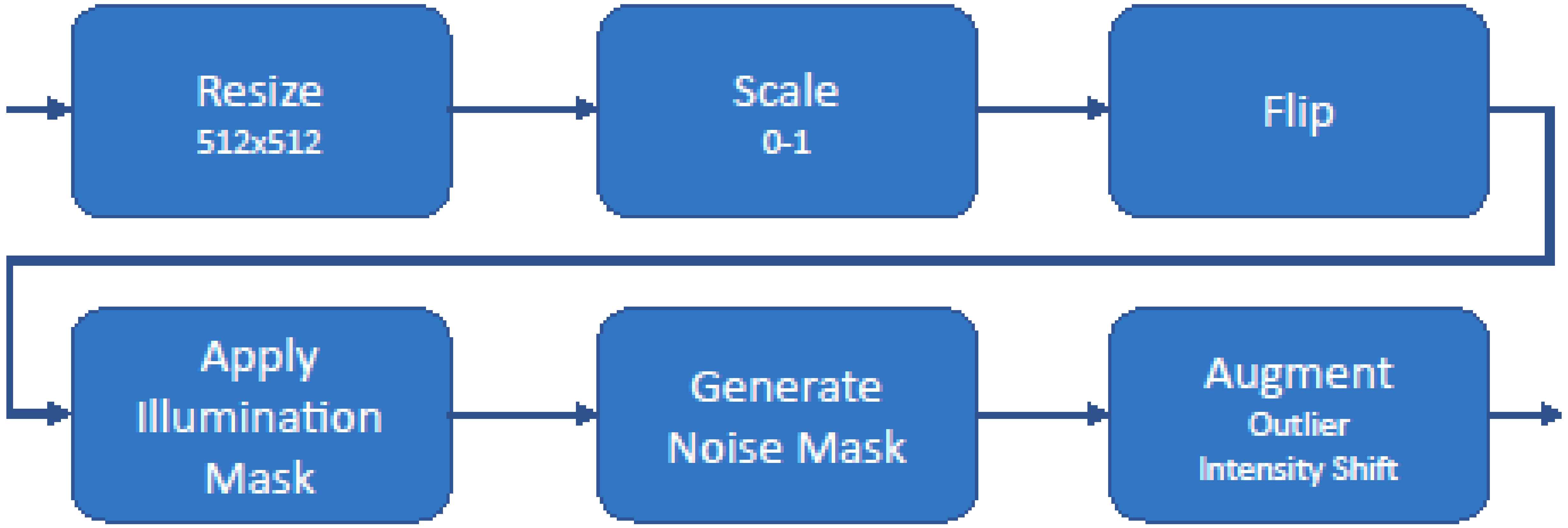
- Training:
  - NYU Depth V2 (indoor, Kinect v1)
  - Added own synthetic holes
- Evaluation
  - NYUV2
  - SceneNet RGB-D (synthetic indoor scenes, low resolution, using depth only, added holes)
  - Self-recorded Azure Kinect data

NYUV2, color/depth [Silberman12]

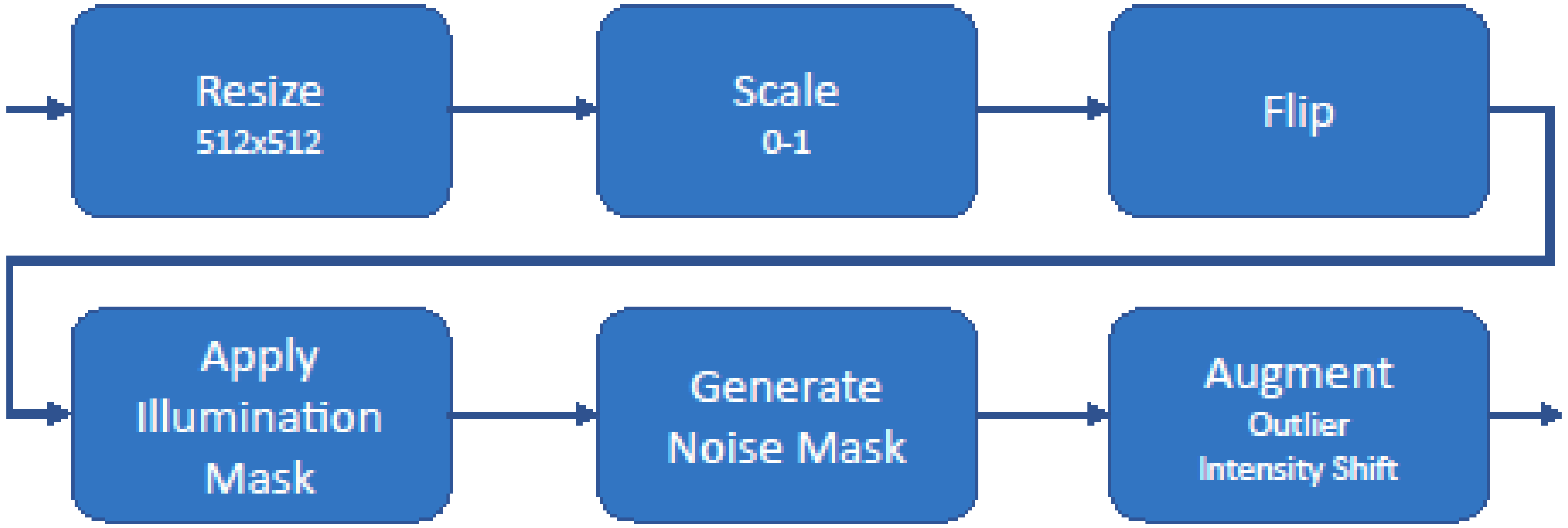


SceneNet, color/depth [McCormac16]

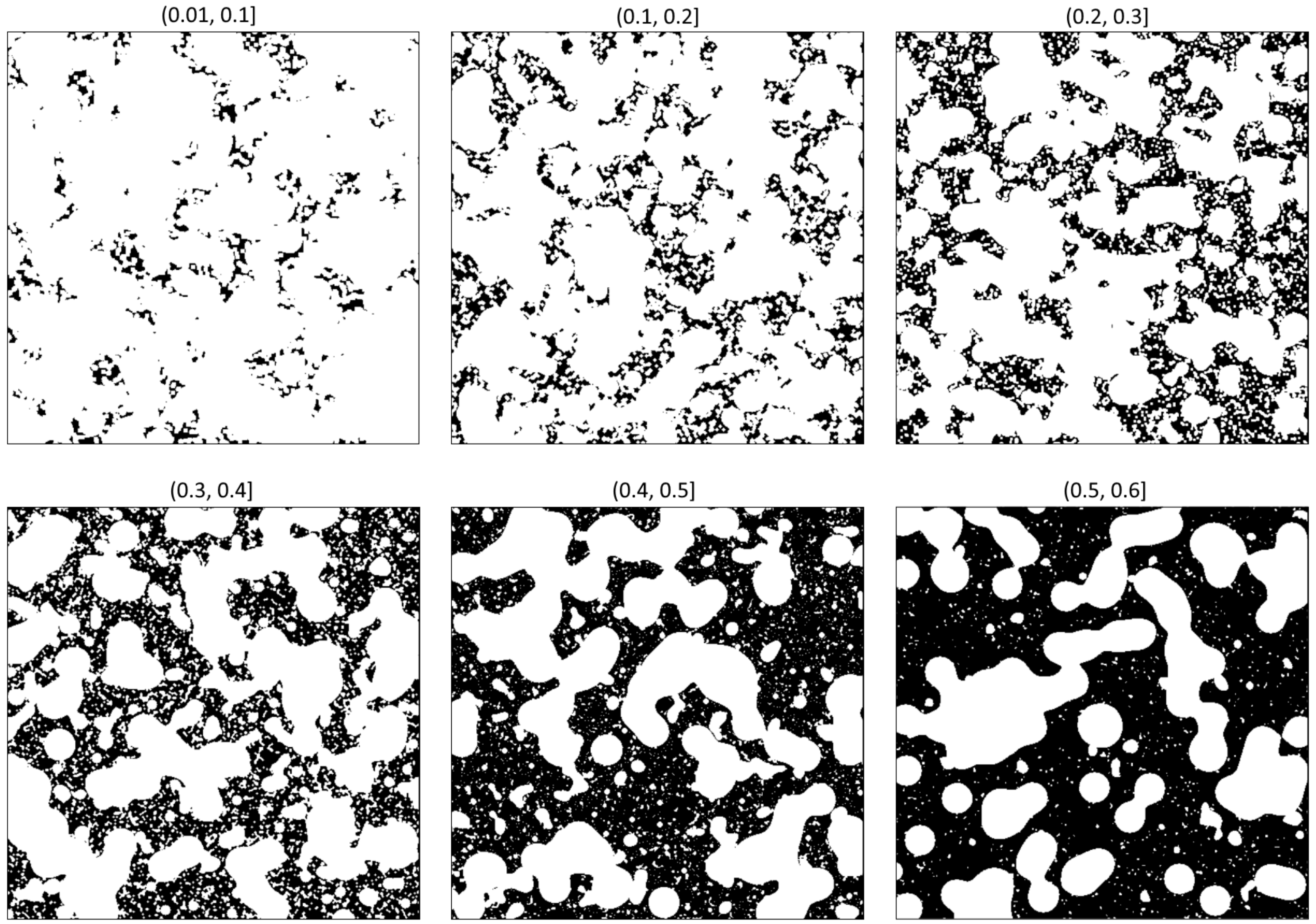
# Preprocessing



# Preprocessing




## Mask Categories

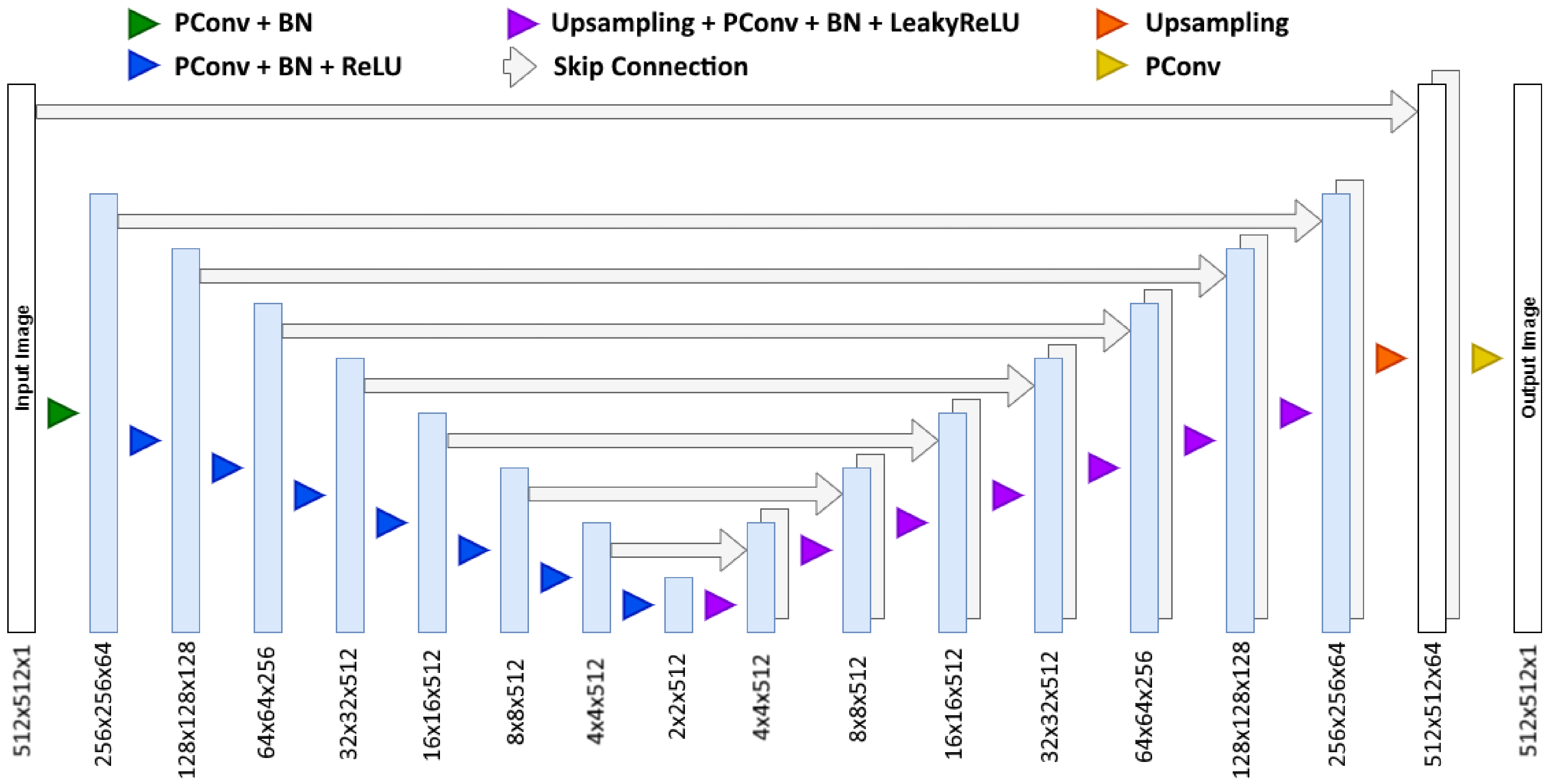



Black: Holes/masked out

Bremen

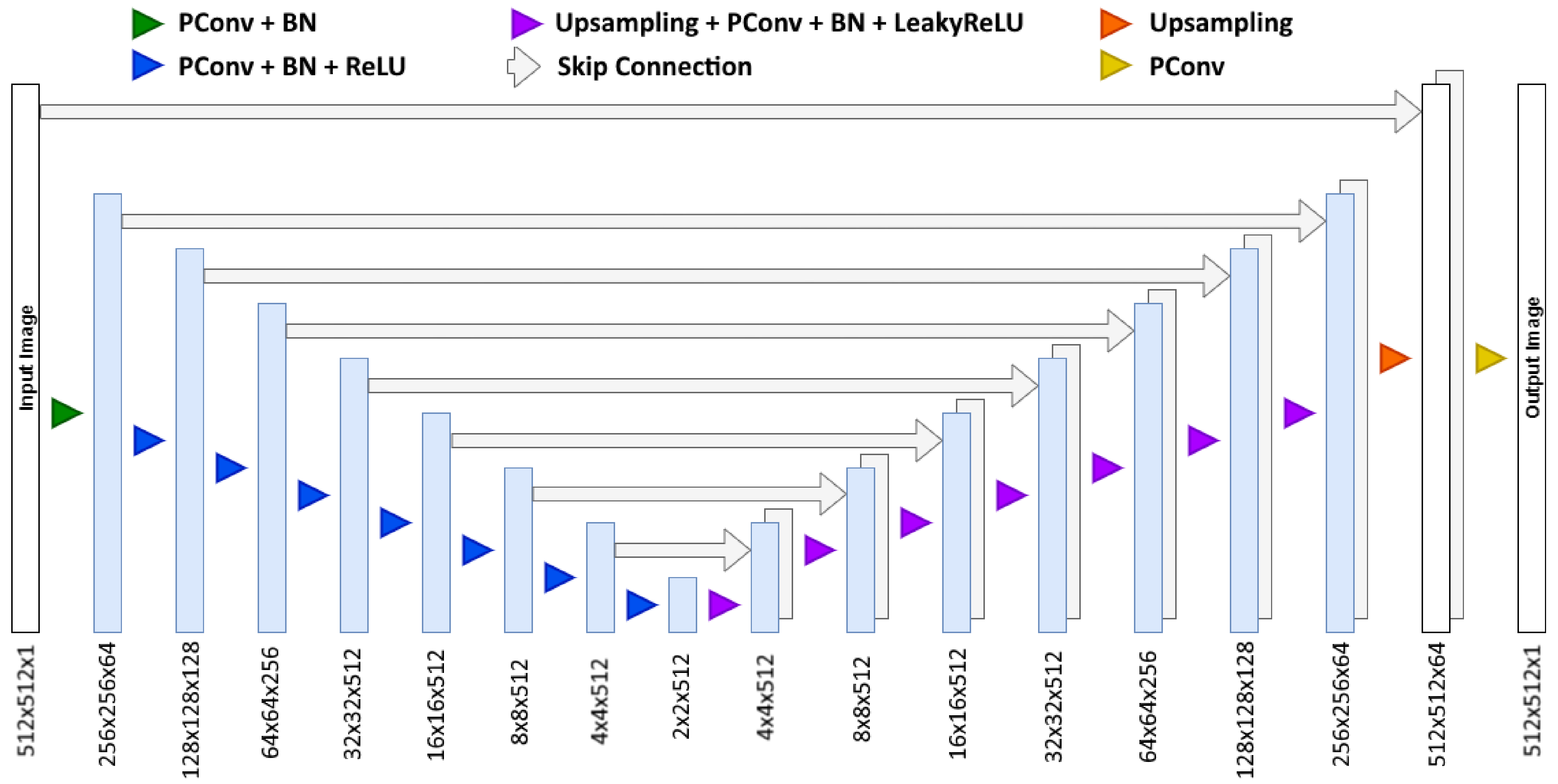
# Network Details

- Partial Convolutionial U-Net [Liu18]



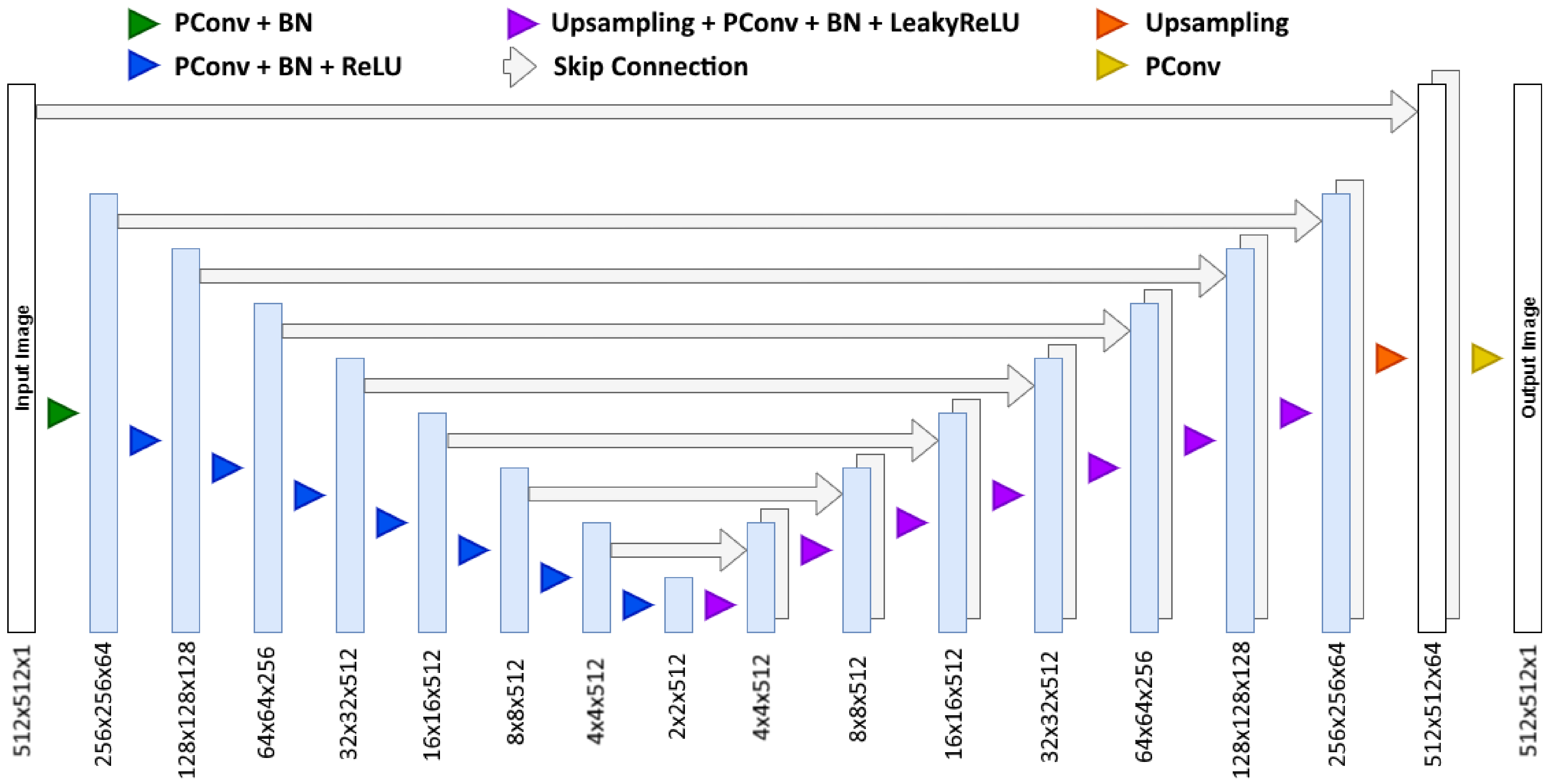
# Network Details

- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers



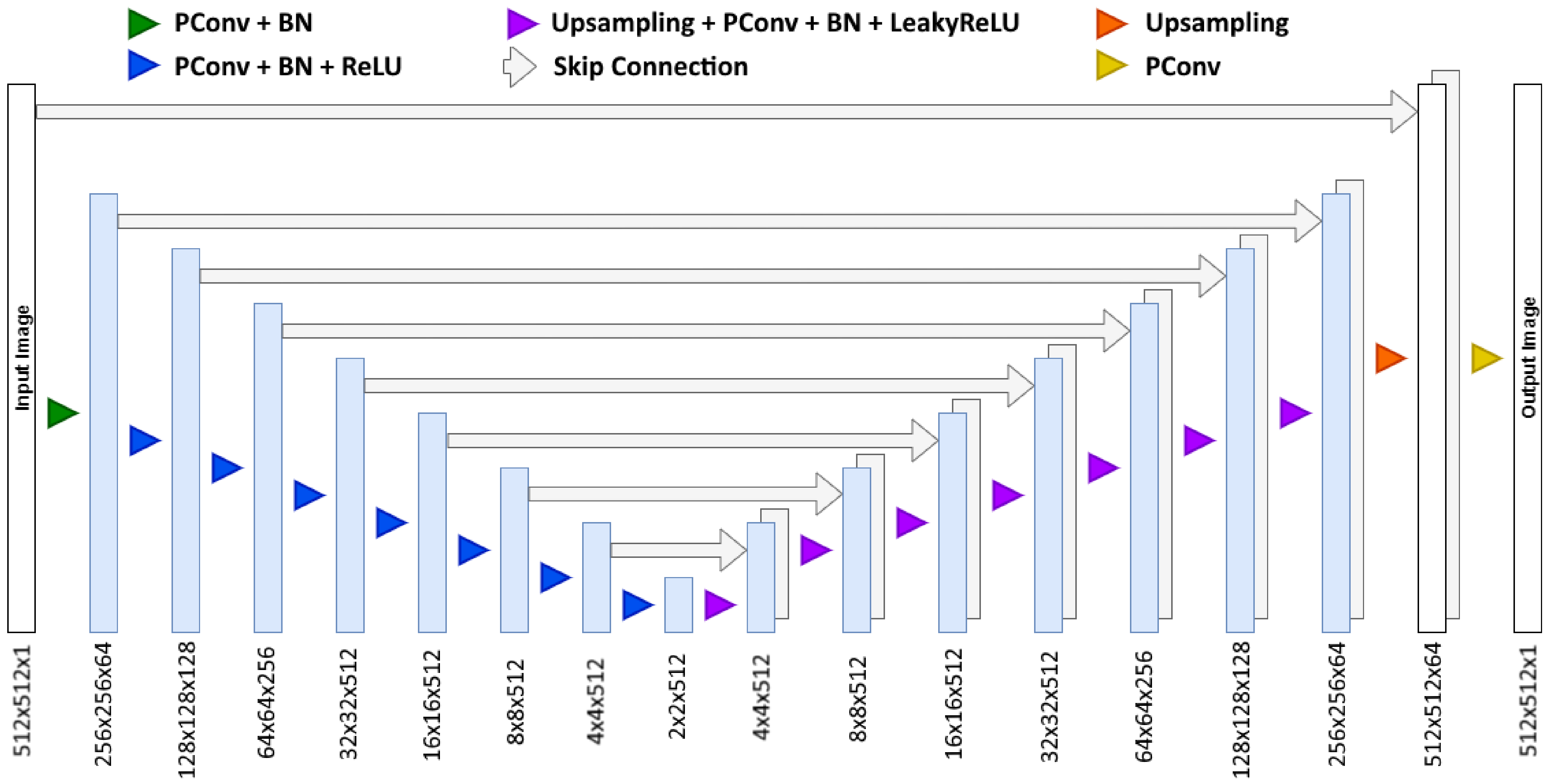
# Network Details

- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers
- Patch-based GAN [Isola17]



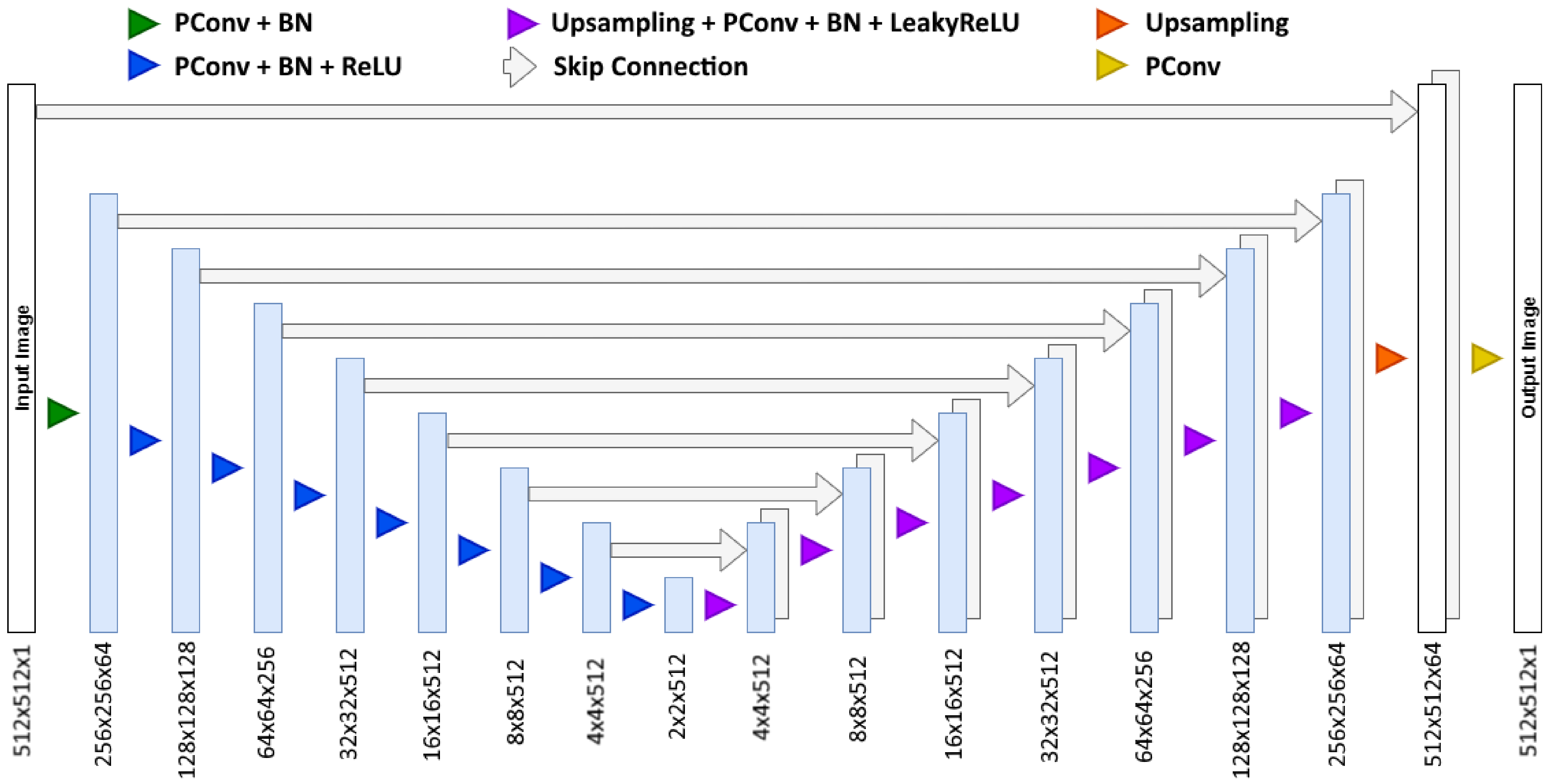
# Network Details

- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers
- Patch-based GAN [Isola17]
  - U-Net generator, convolutional PatchGAN classifier as discriminator



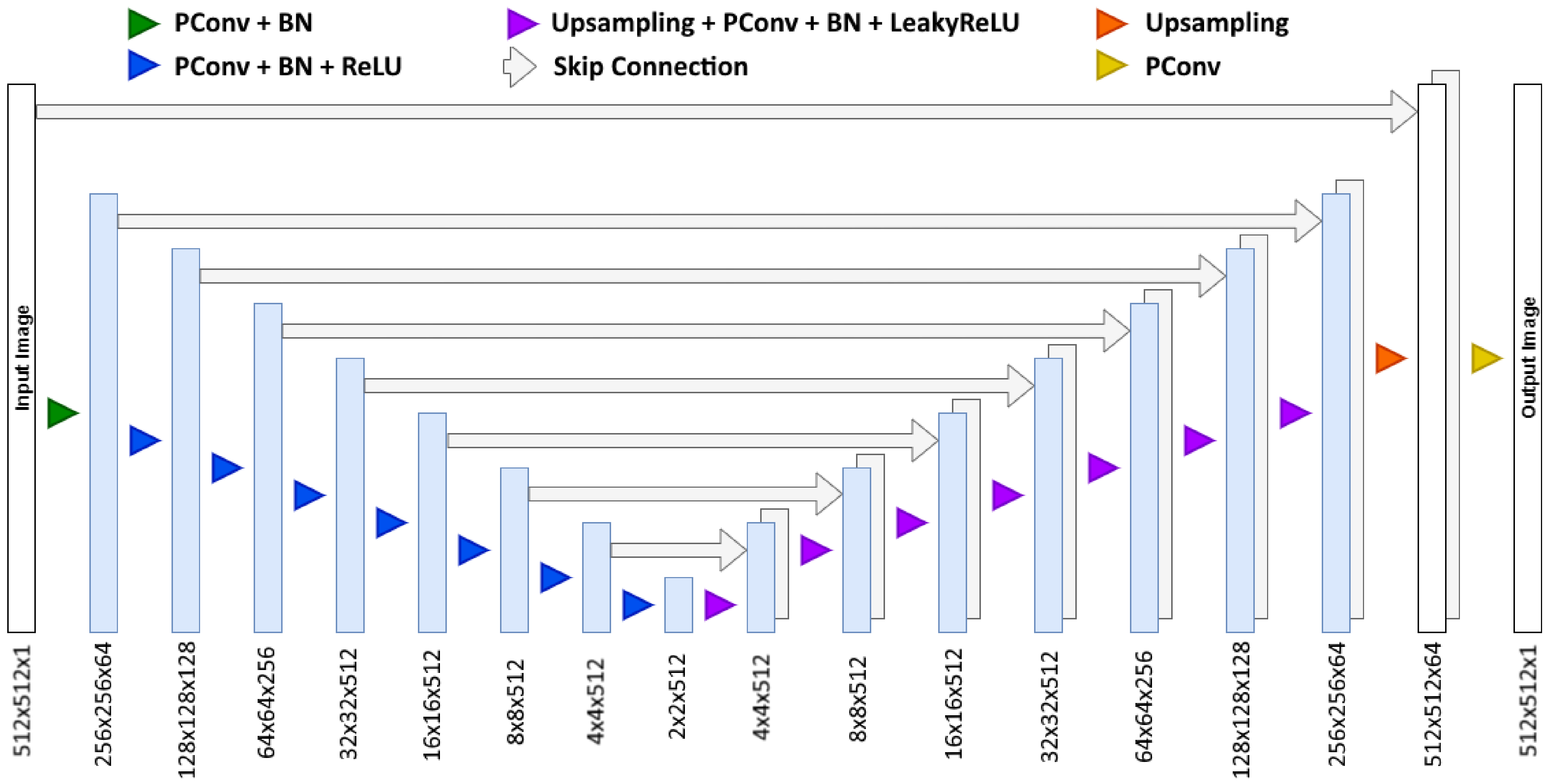
# Network Details

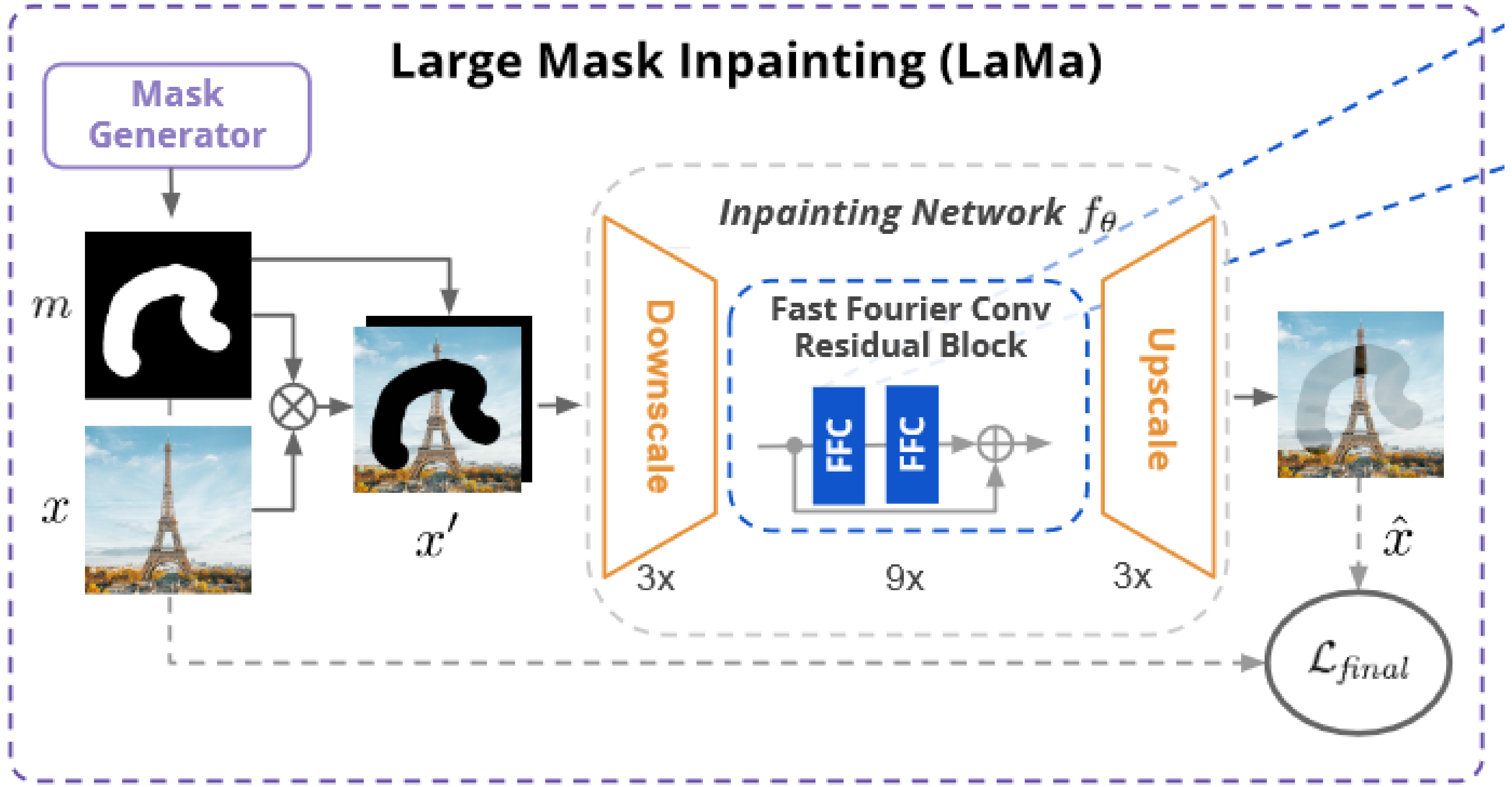
- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers
- Patch-based GAN [Isola17]
  - U-Net generator, convolutional PatchGAN classifier as discriminator
- Standard U-Net



# Network Details

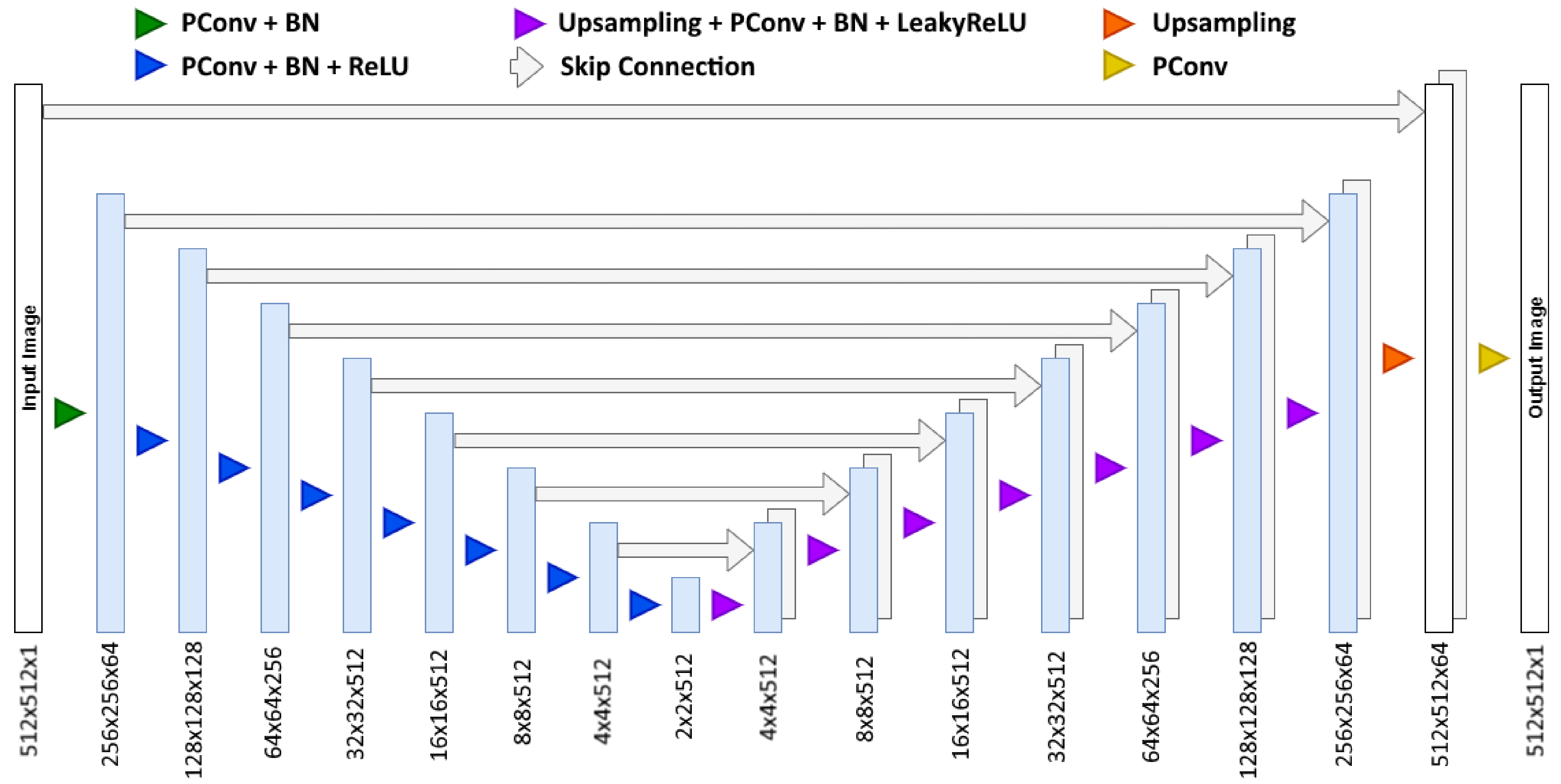
- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers
- Patch-based GAN [Isola17]
  - U-Net generator, convolutional PatchGAN classifier as discriminator
- Standard U-Net
- LaMa [Suvorov22]

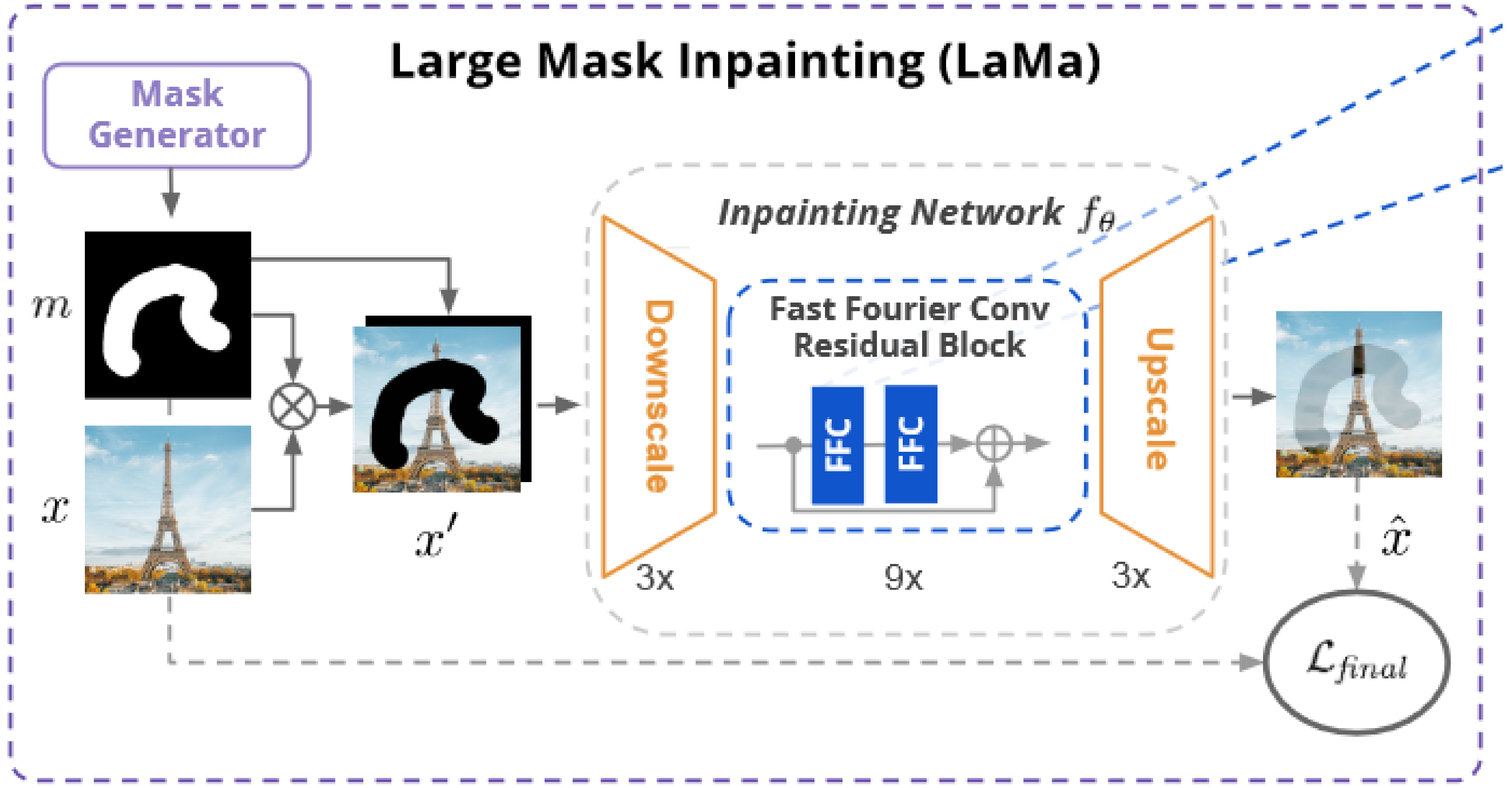


# Network Details

- Partial Convolutionial U-Net [Liu18]
  - Convolutions masked on valid pixels
  - Dynamic mask updates between layers
- Patch-based GAN [Isola17]
  - U-Net generator, convolutional PatchGAN classifier as discriminator
- Standard U-Net
- LaMa [Suvorov22]
  - Fourier convolutions provide large receptive field
  - Large training masks



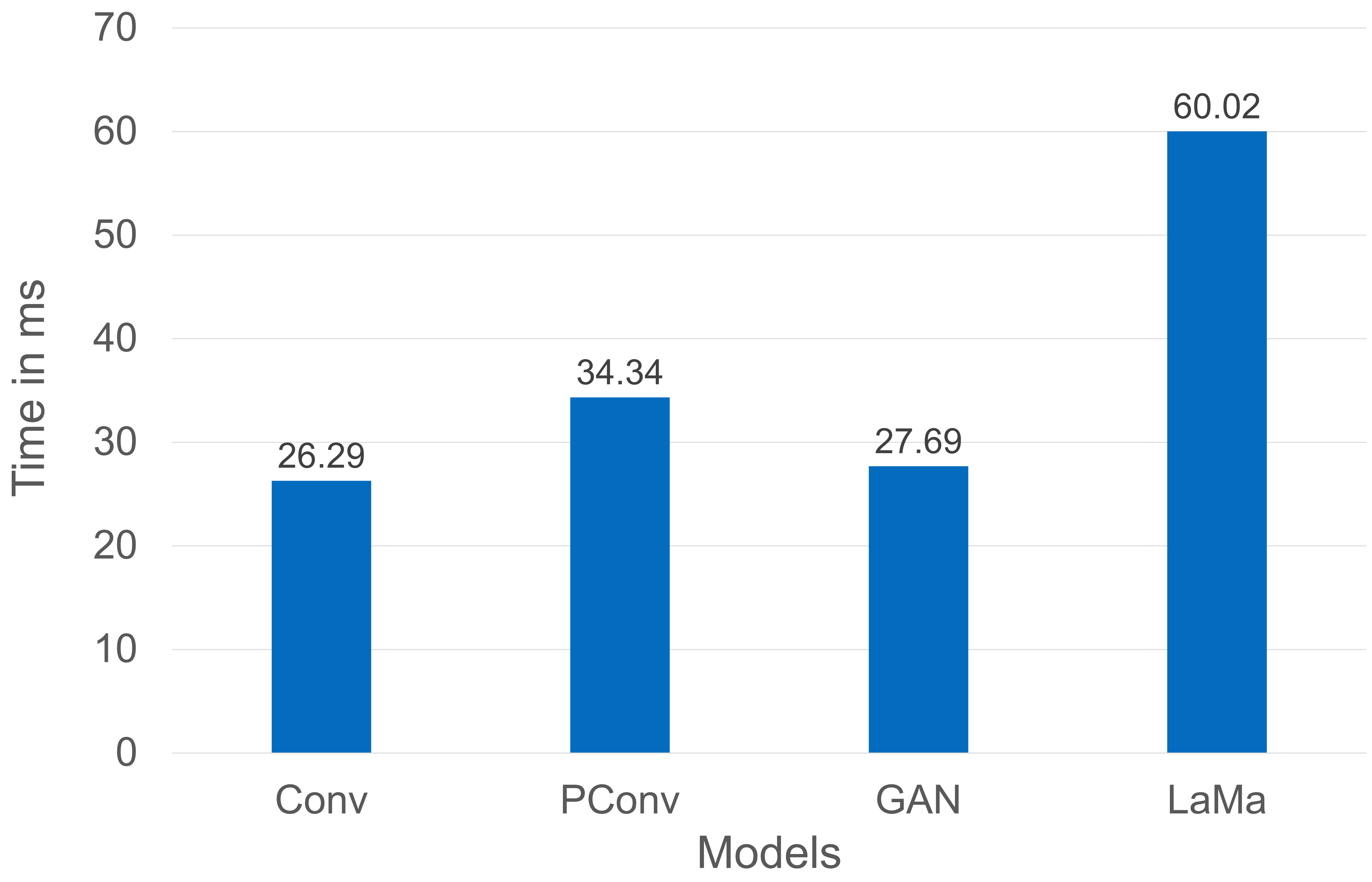
# Training Procedure & Loss Functions

- Trained for 7 epochs (LaMa: 5), batch size 2 (LaMa: 5)
- Losses:
  - Conv/PConv (like the original paper):

    Two per-pixel accuracy losses, a perceptual loss, two style losses, a total variation loss
  - GAN: Combination of above with original generator loss (including L1 loss)
  - LaMa (like the original paper for comparability):

    A high receptive field perceptual loss, an adversarial loss, a discriminator-based perceptual loss, and gradient penalty

# Results - Inference Timings

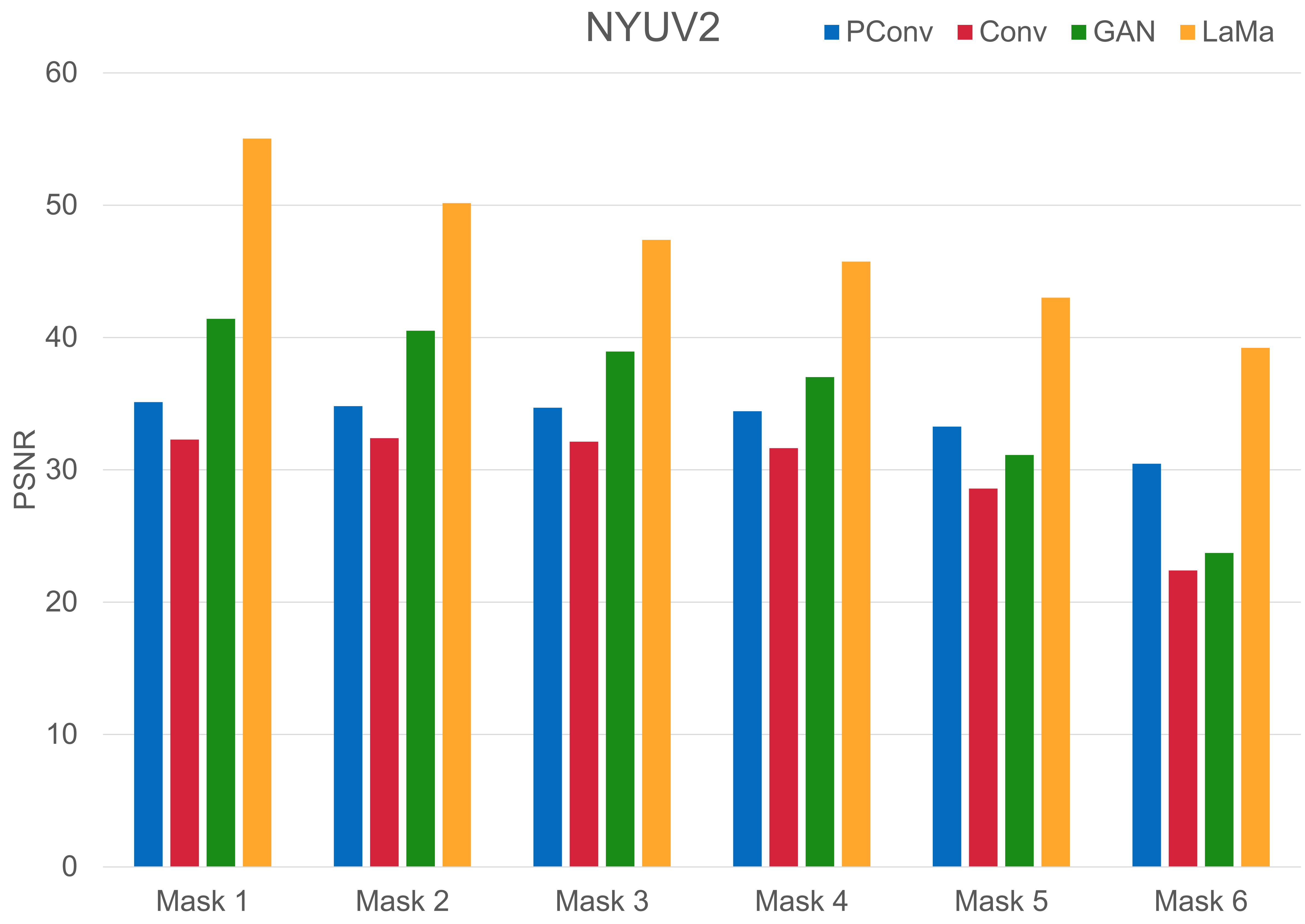


# Results - Quantitative Analysis

- Measured metrics: MAE, MSE, PSNR, SSIM

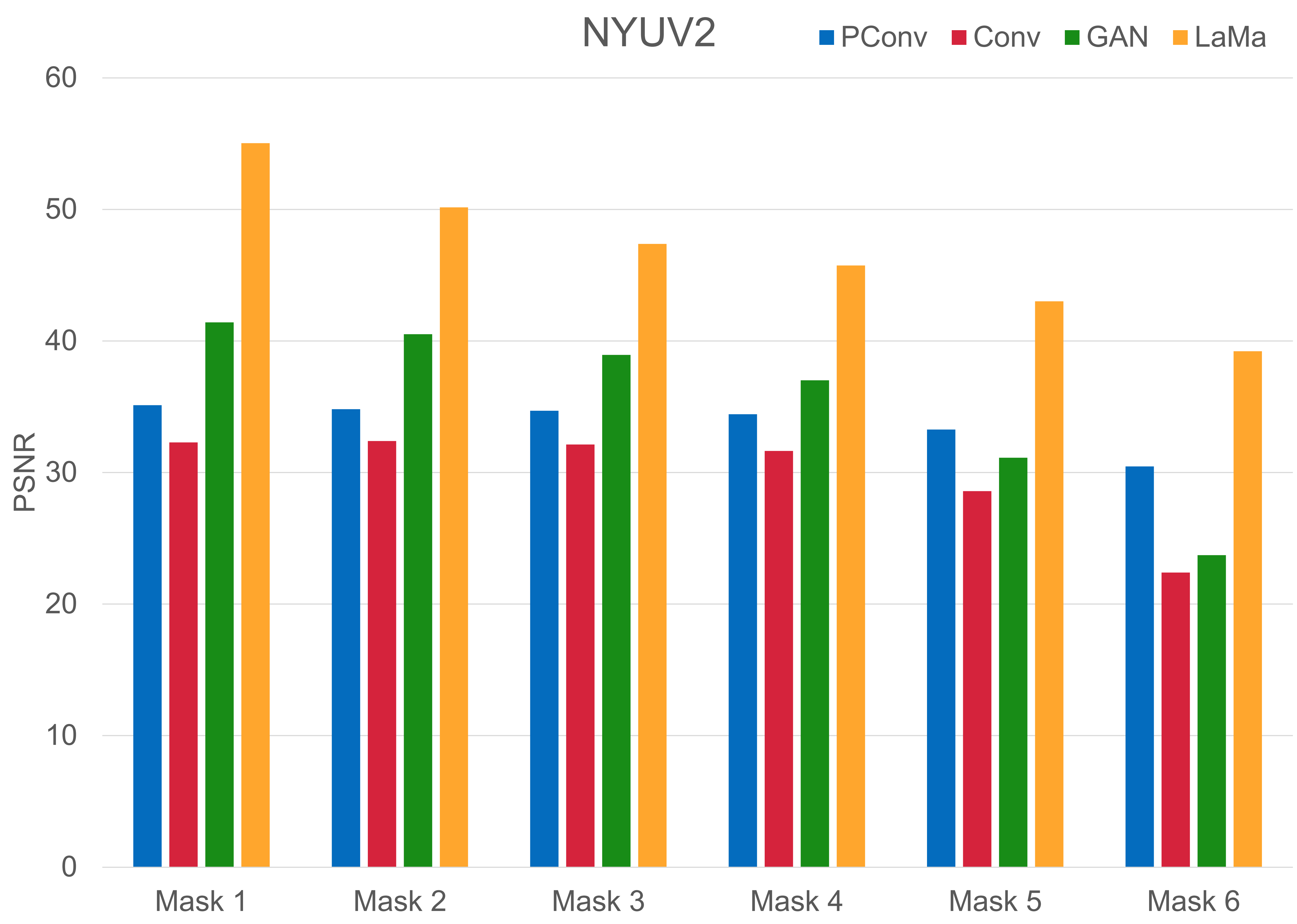
# Results - Quantitative Analysis

- Measured metrics: MAE, MSE, PSNR, SSIM



# Results - Quantitative Analysis

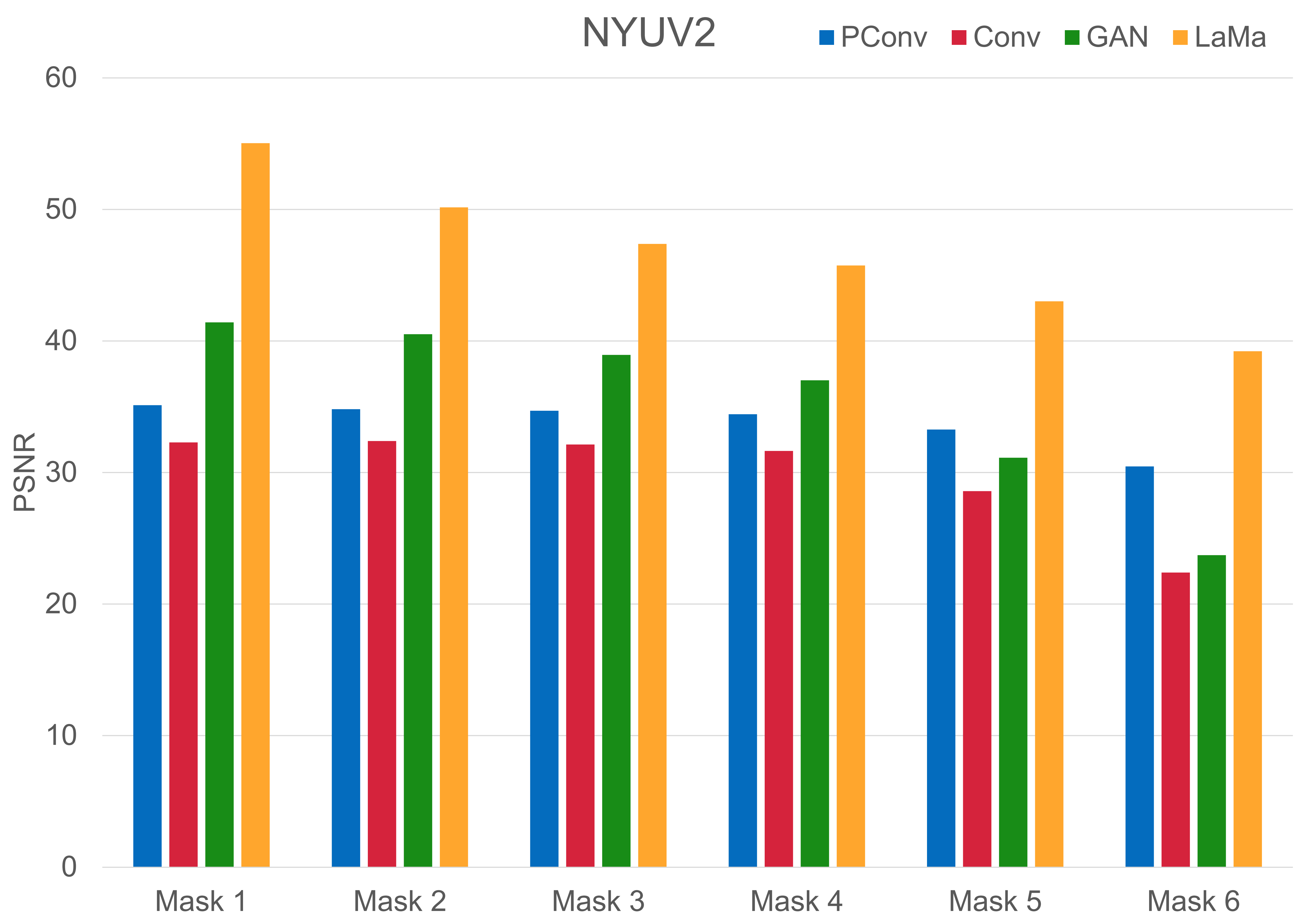
- Measured metrics: MAE, MSE, PSNR, SSIM

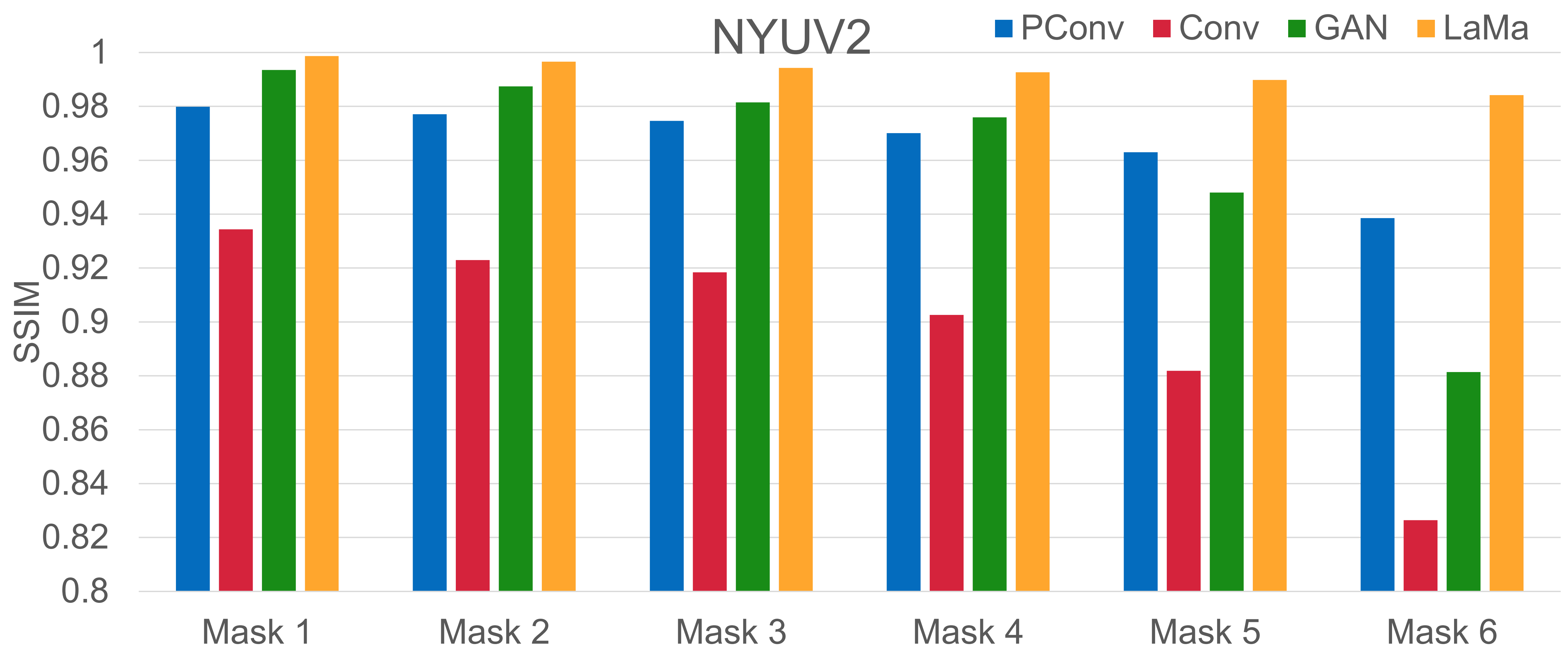



LaMa best, GAN second best on small/medium masks, PConv on larger ones and most consistent

# Results - Quantitative Analysis

- Measured metrics: MAE, MSE, PSNR, SSIM

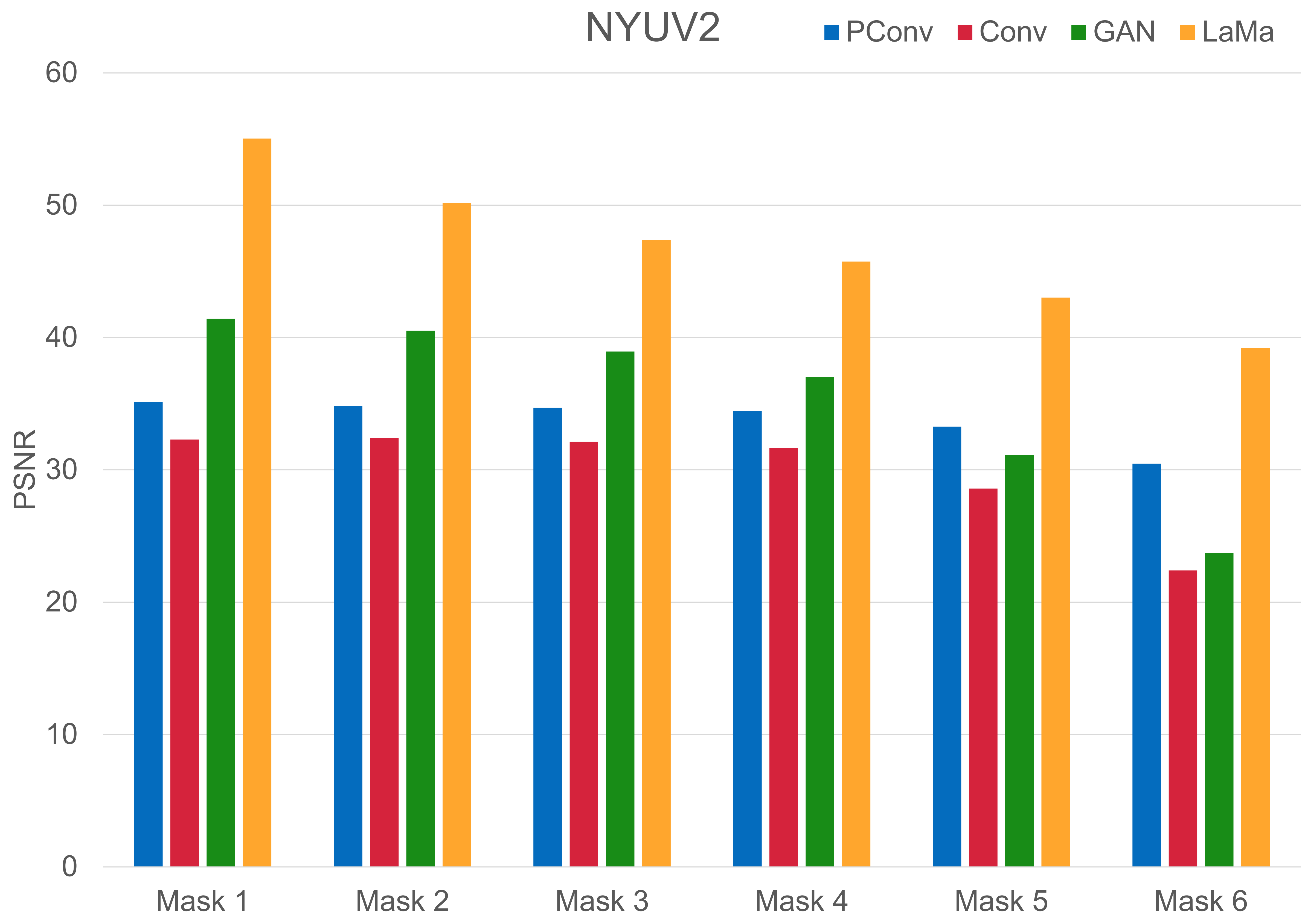


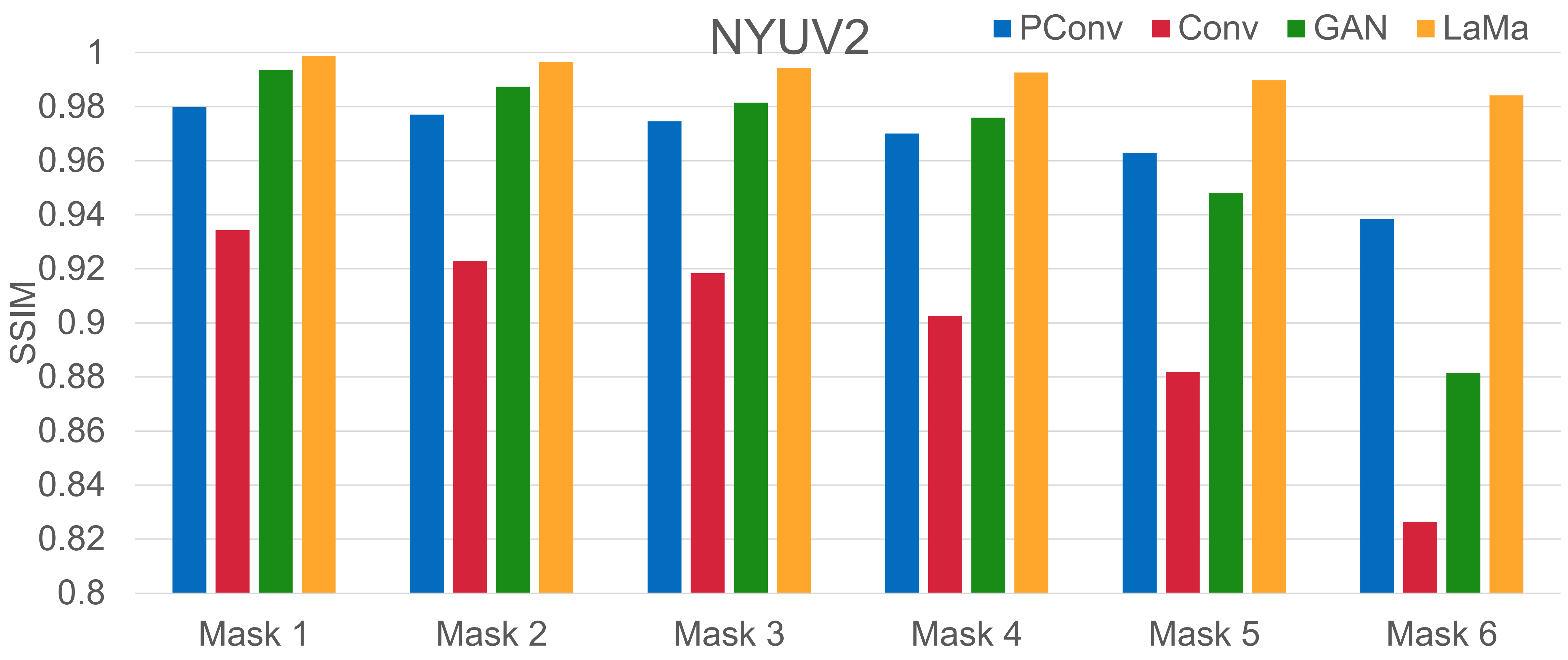



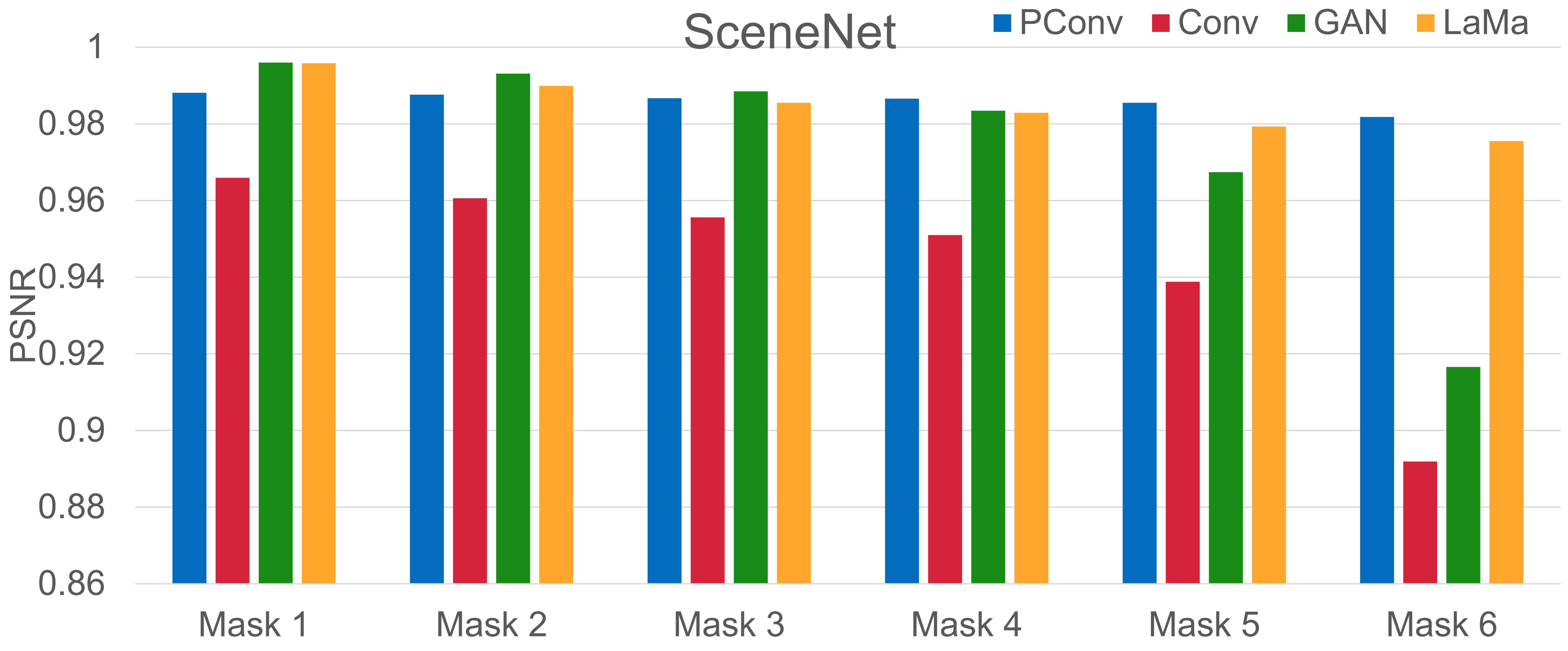

LaMa best, GAN second best on small/medium masks, PConv on larger ones and most consistent

# Results - Quantitative Analysis

- Measured metrics: MAE, MSE, PSNR, SSIM

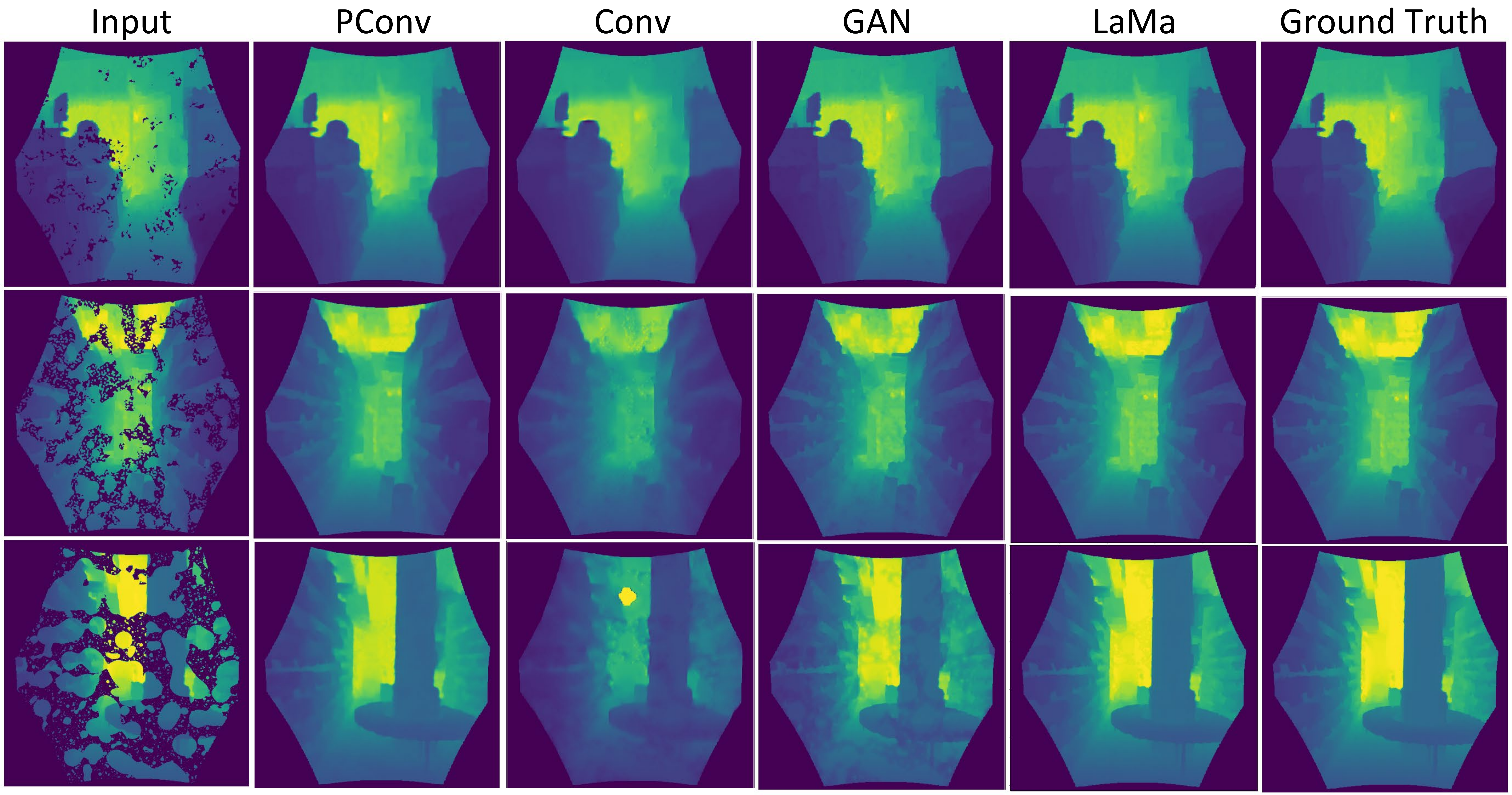


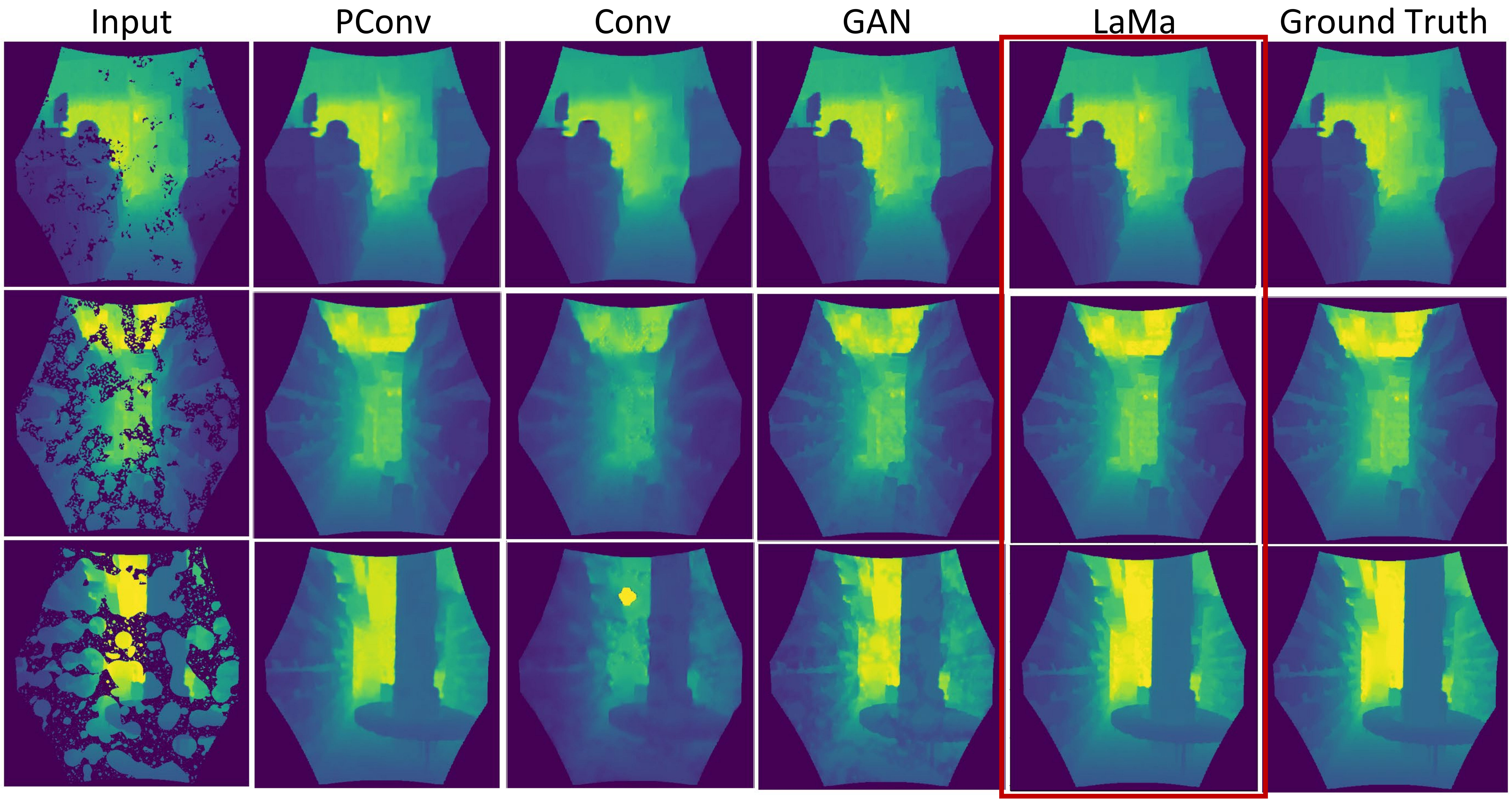




LaMa best, GAN second best on small/medium masks, PConv on larger ones and most consistent

# Results - Qualitative Comparision NYUV2

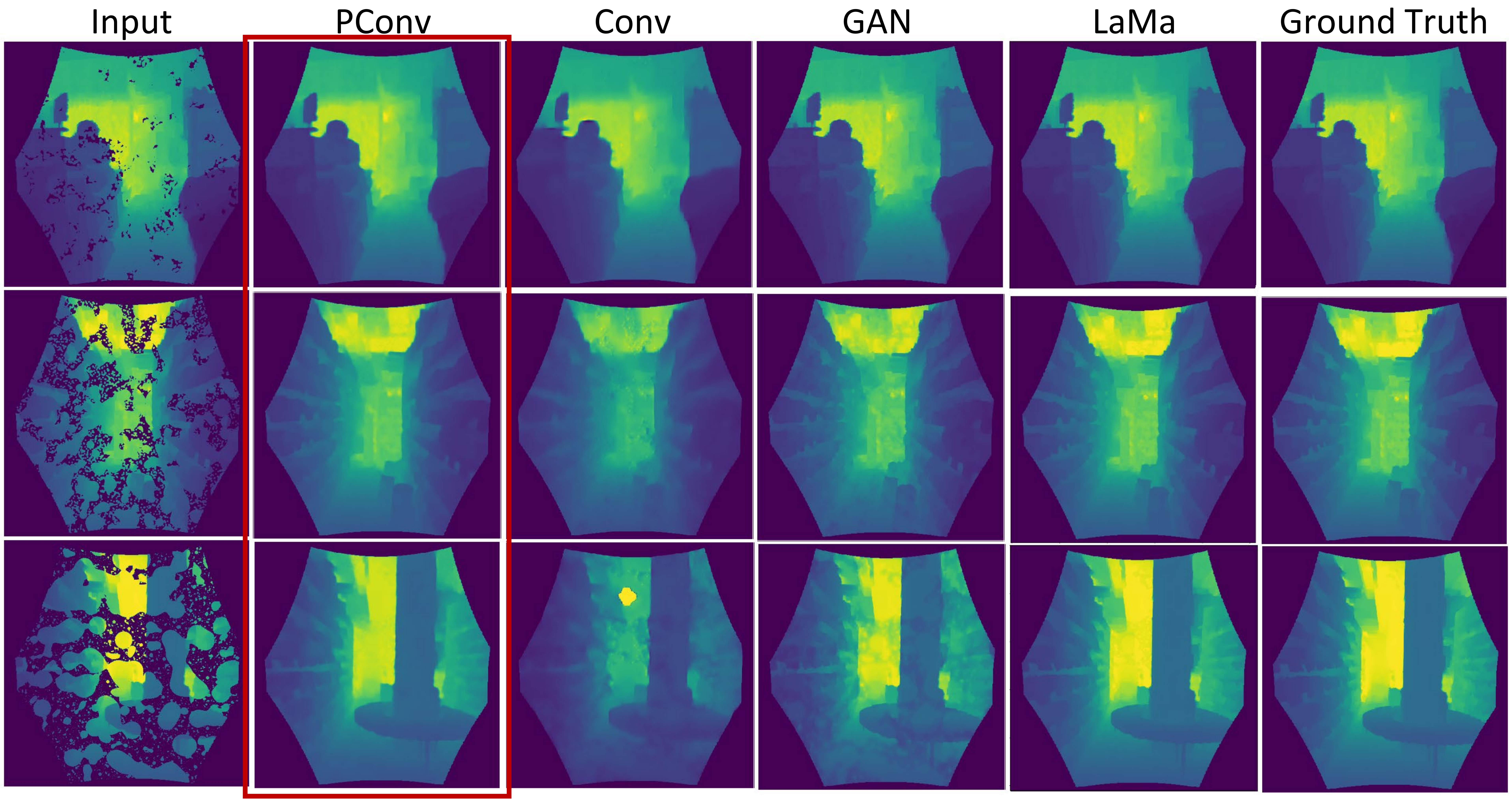


# Results - Qualitative Comparision NYUV2



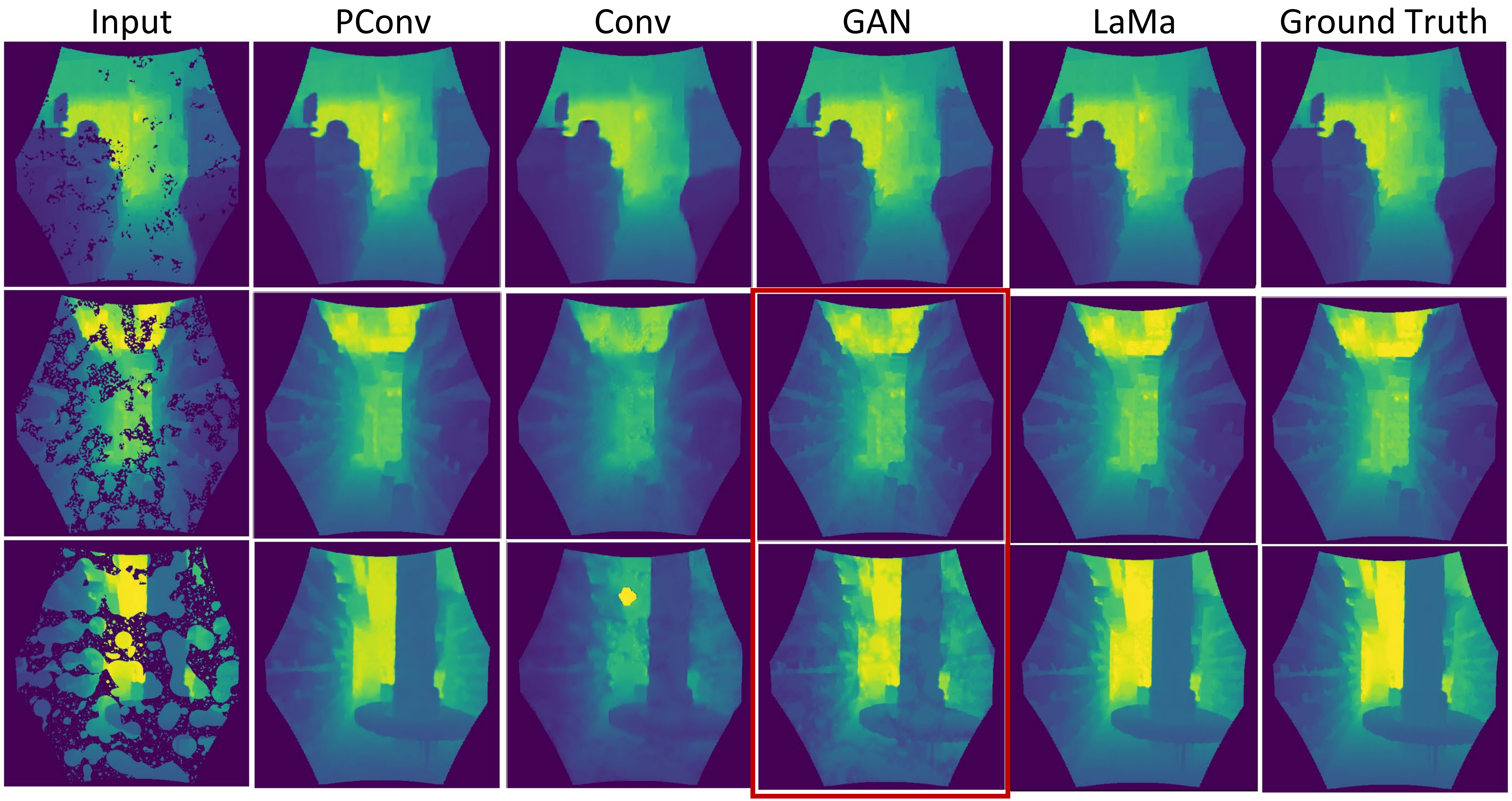

Lama perform best,

# Results - Qualitative Comparision NYUV2



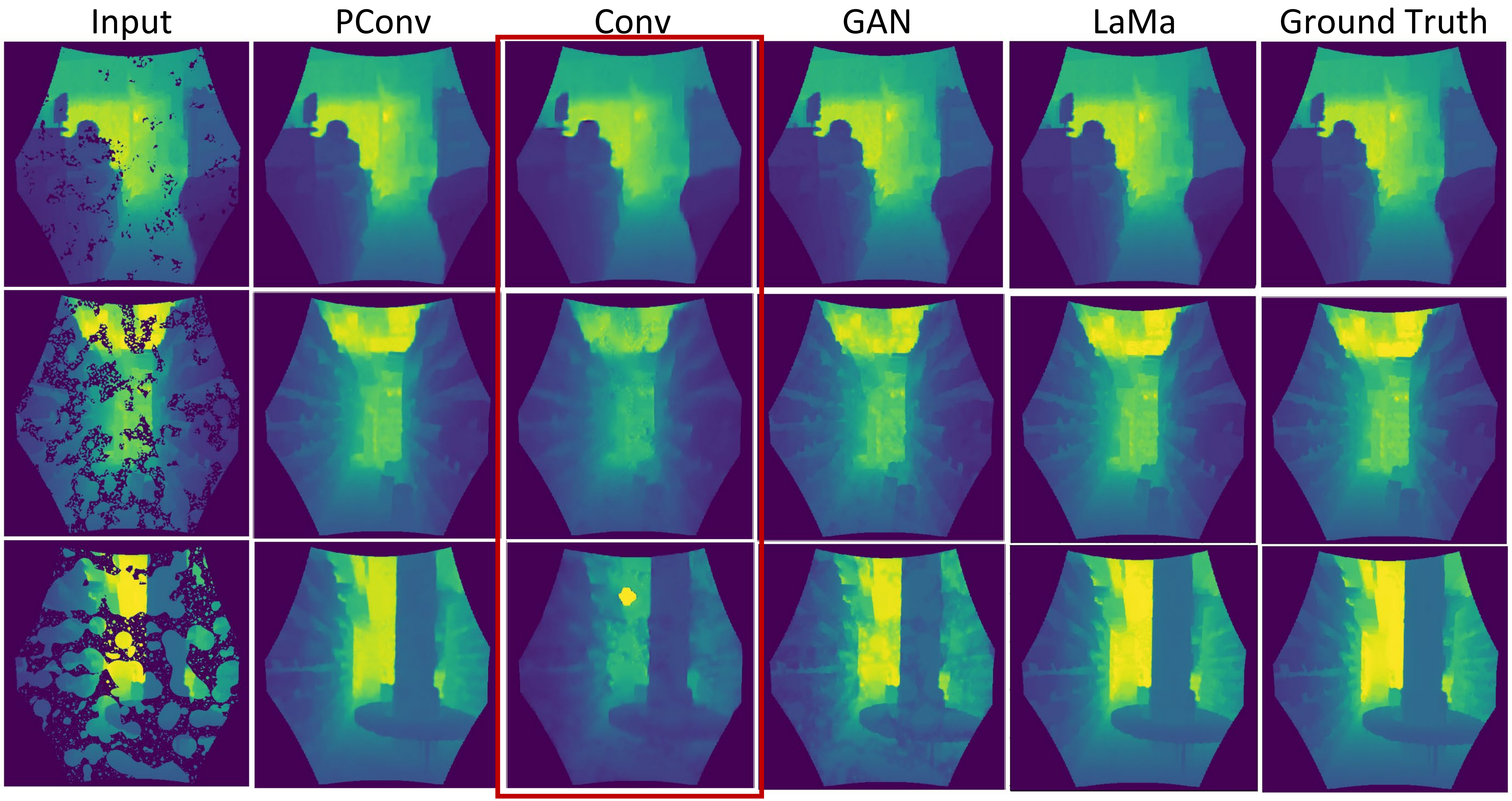

Lama perform best, PConv second best,

# Results - Qualitative Comparision NYUV2



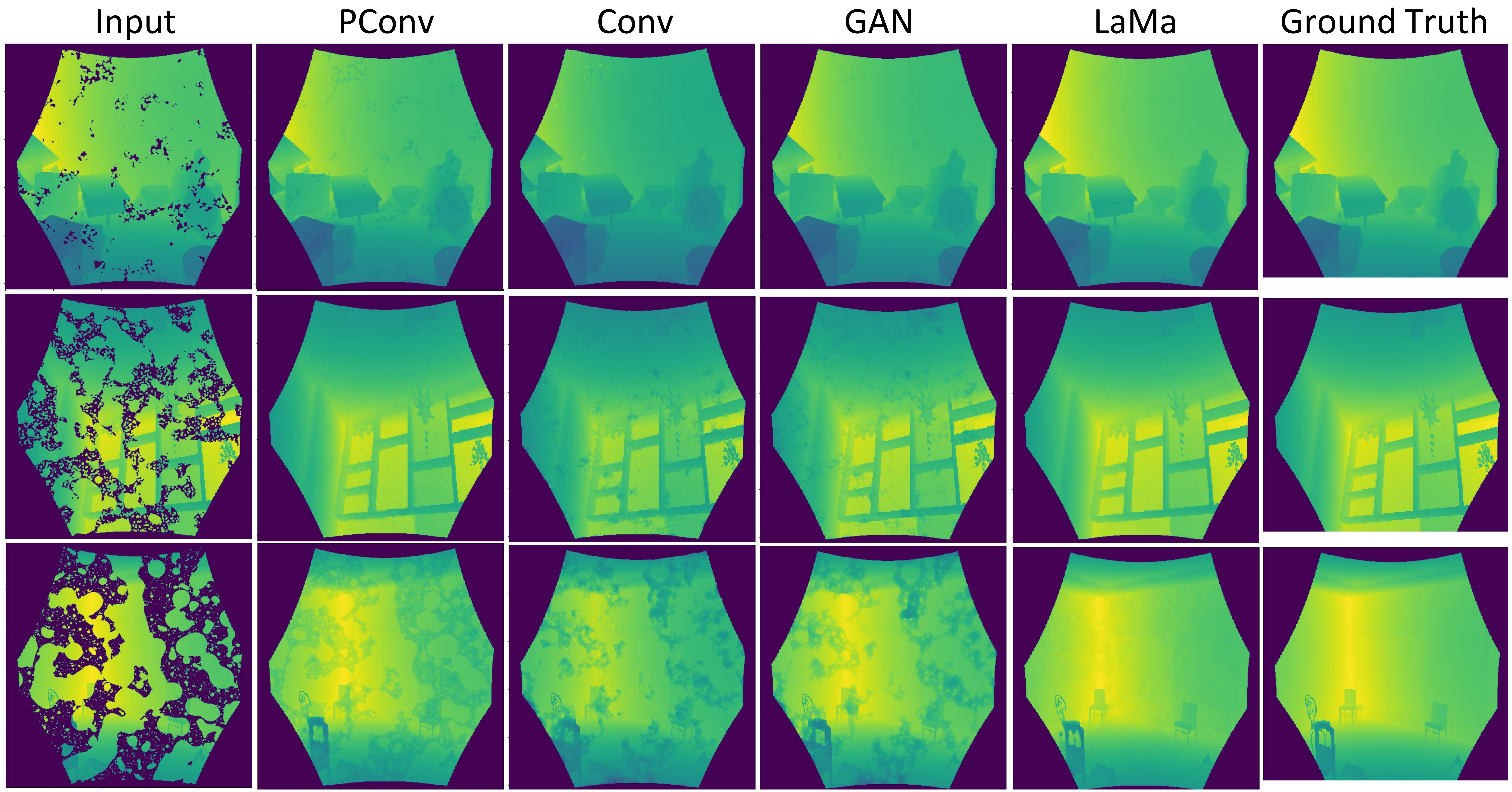

Lama perform best, PConv second best, GAN with issues on larger masks,

# Results - Qualitative Comparision NYUV2

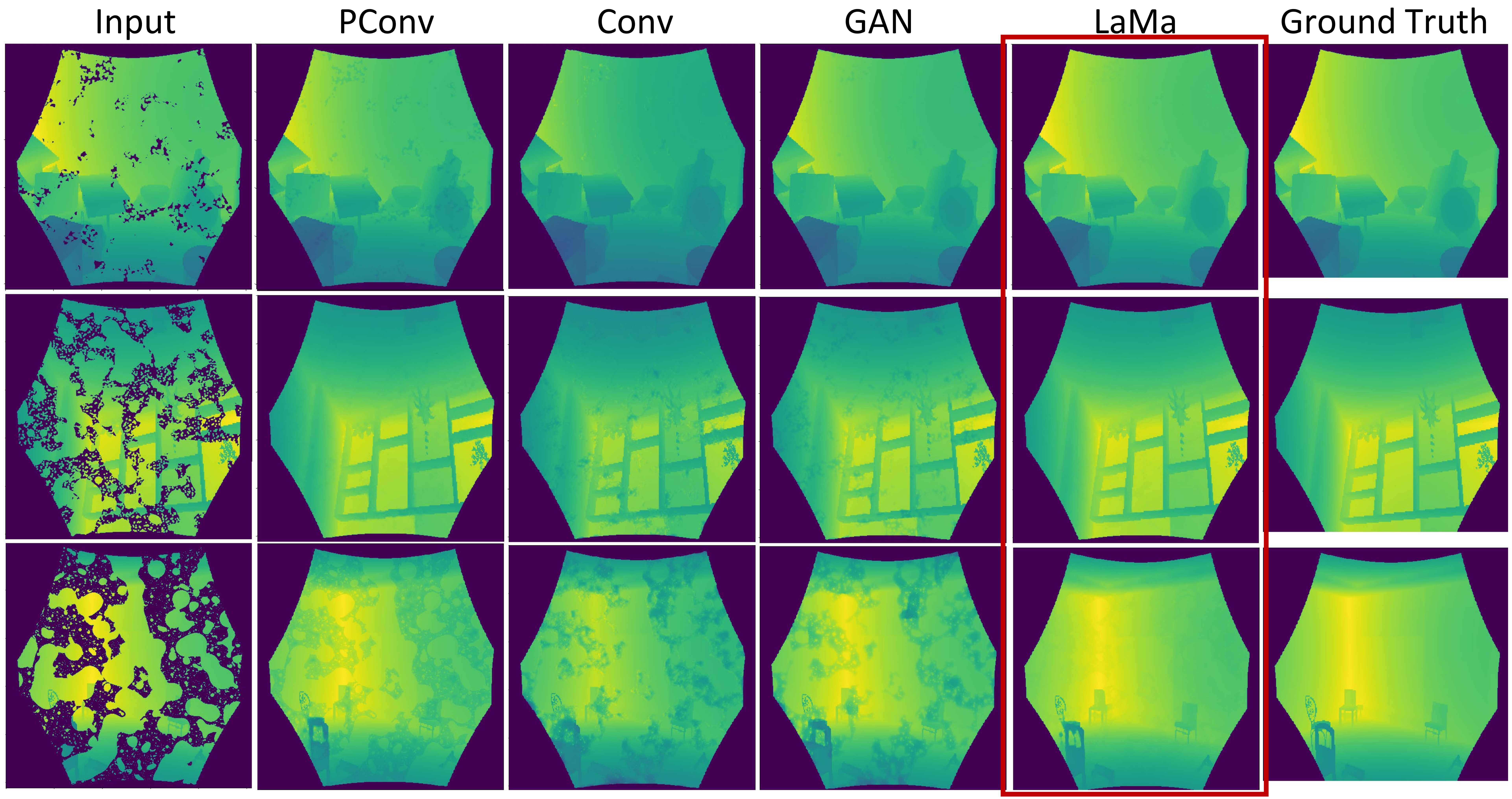



Lama perform best, PConv second best, GAN with issues on larger masks, Conv worst

# Results - Qualitative Comparision SceneNet

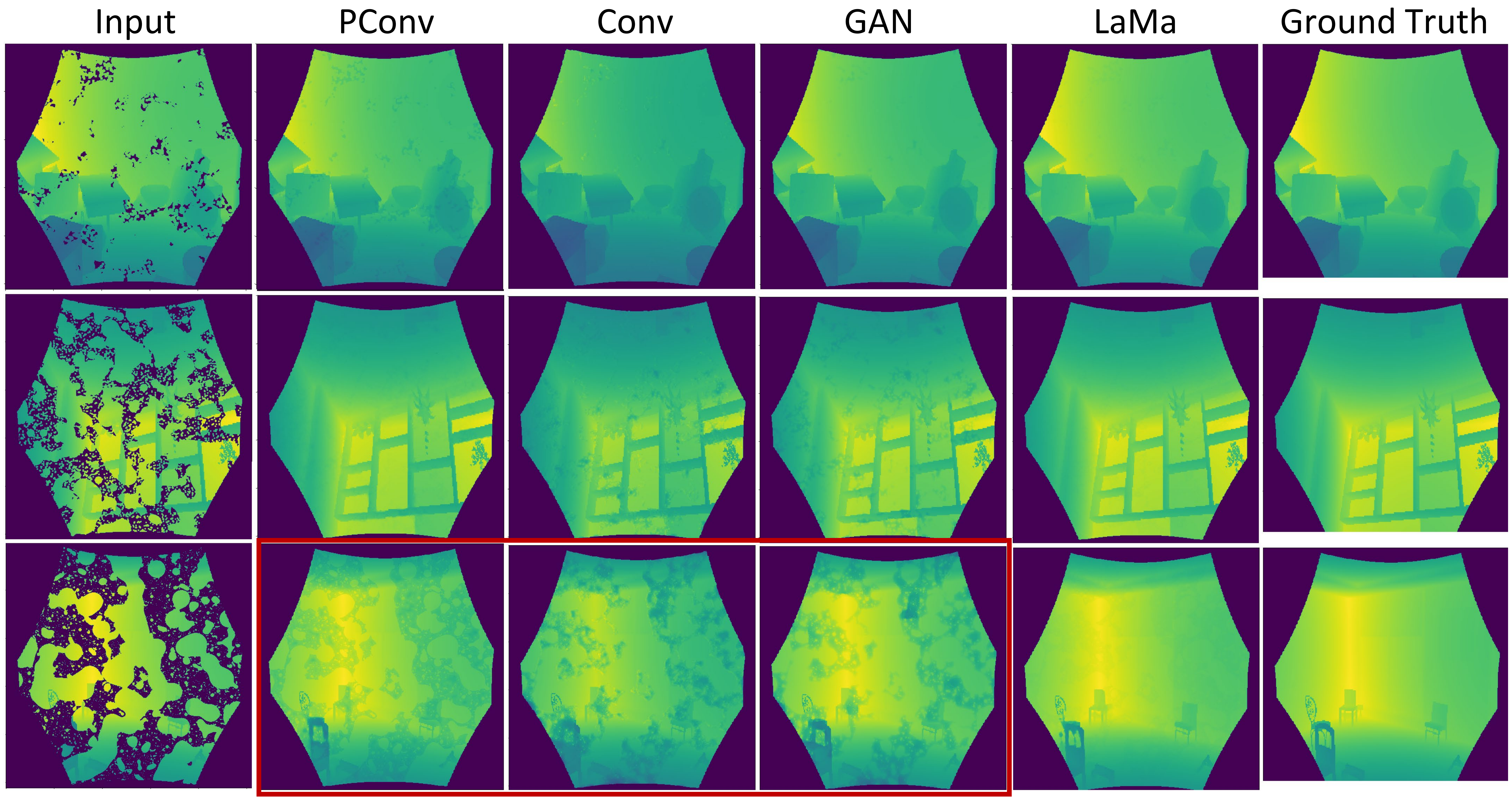


# Results - Qualitative Comparision SceneNet



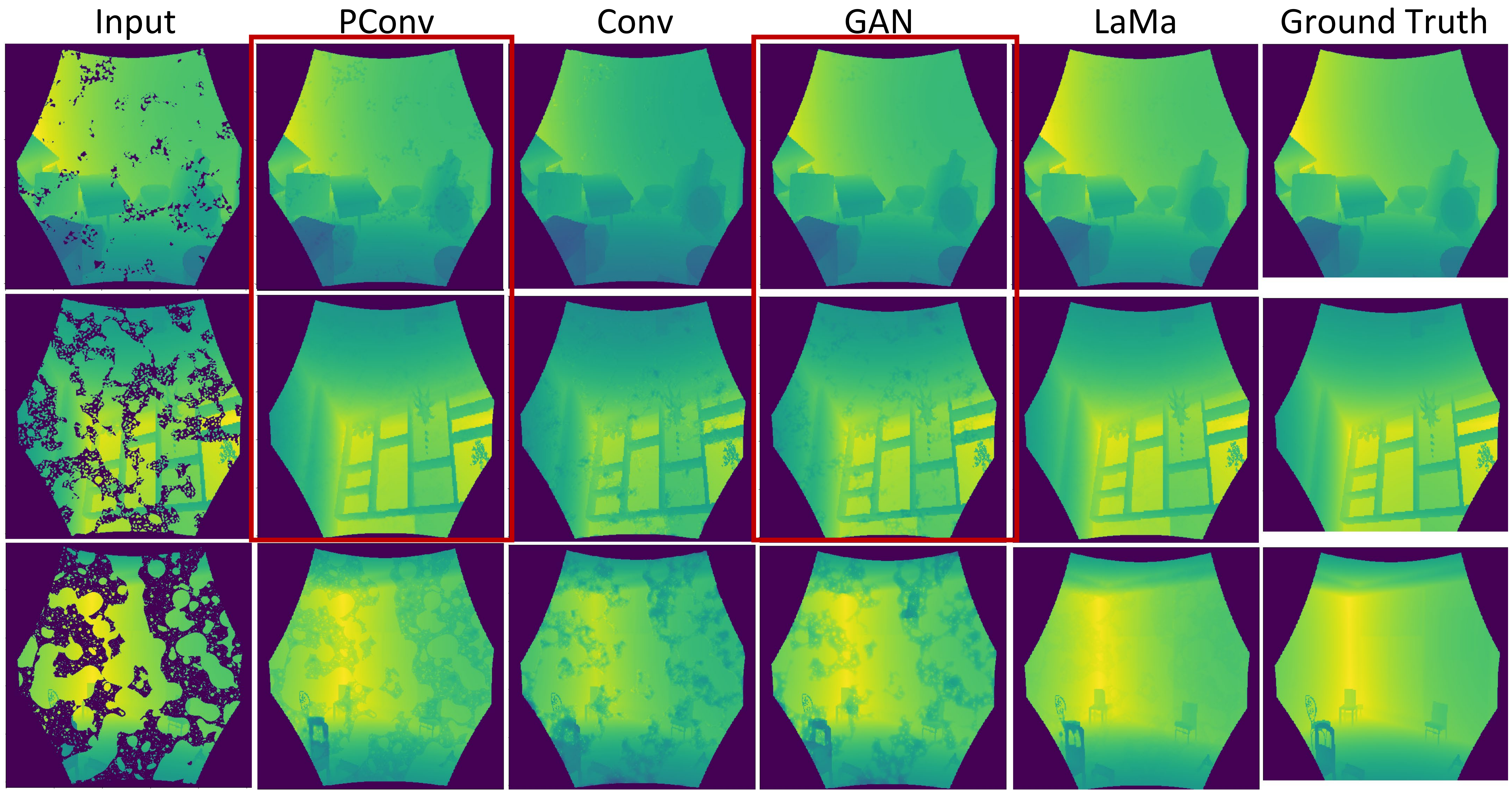

LaMa best again,

# Results - Qualitative Comparision SceneNet



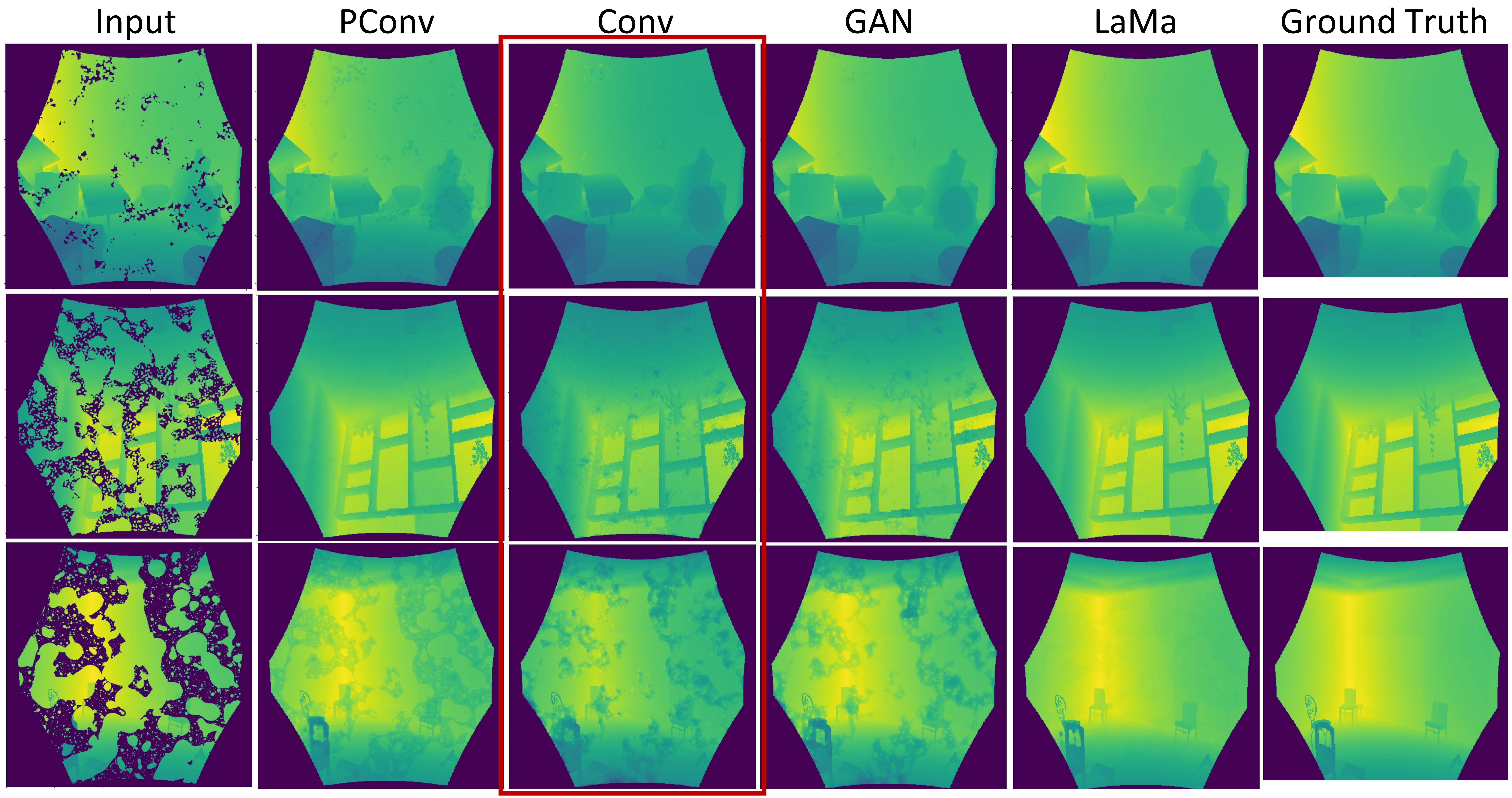

LaMa best again, others artefacts,

# Results - Qualitative Comparision SceneNet




LaMa best again, others artefacts, PConv/GAN ok in medium/small categories,

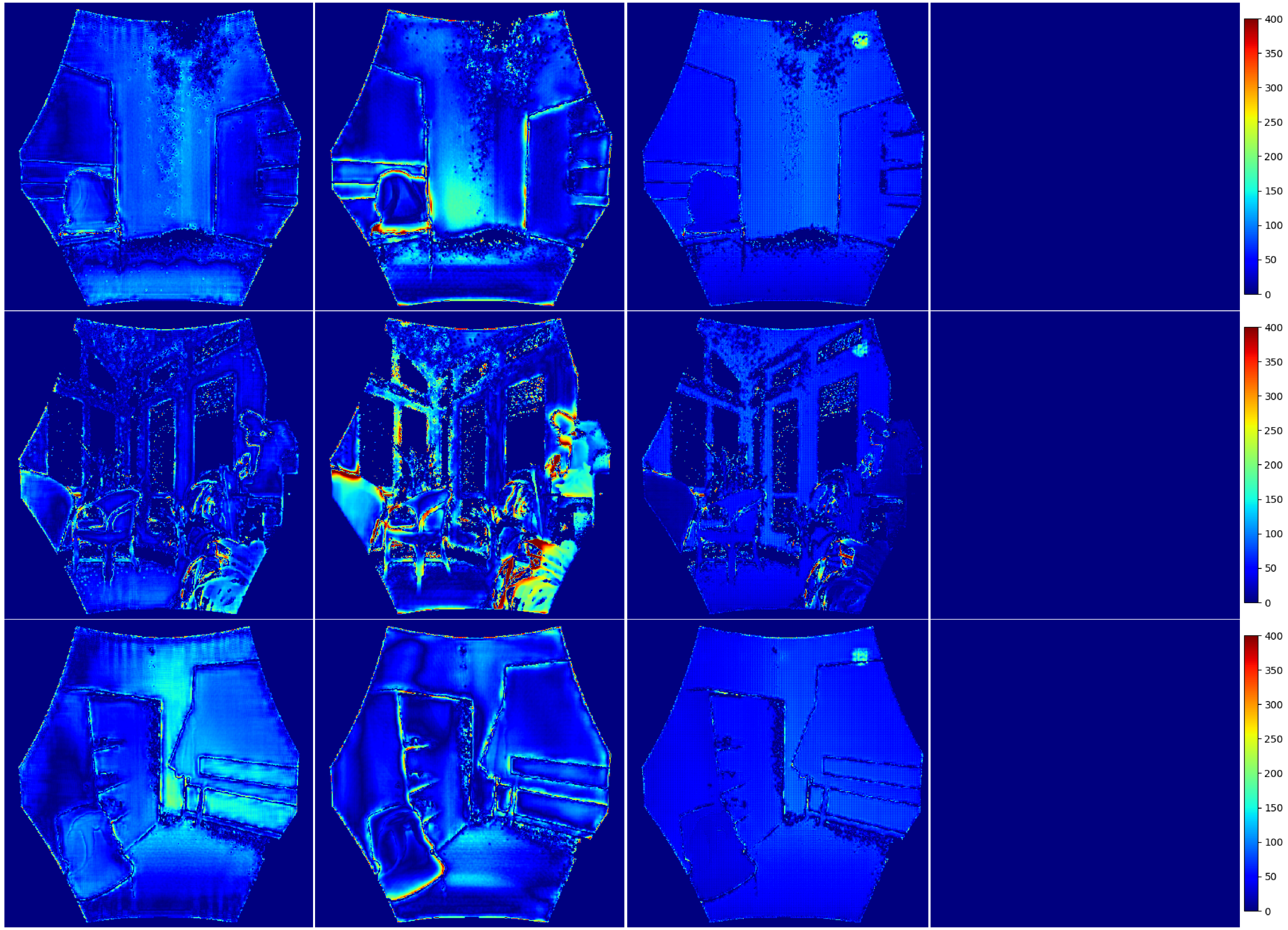
# Results - Qualitative Comparision SceneNet




LaMa best again, others artefacts, PConv/GAN ok in medium/small categories, Conv worst again

# Results - Qualitative Comparision Own dataset

Color-coded deltas of valid areas (less better)

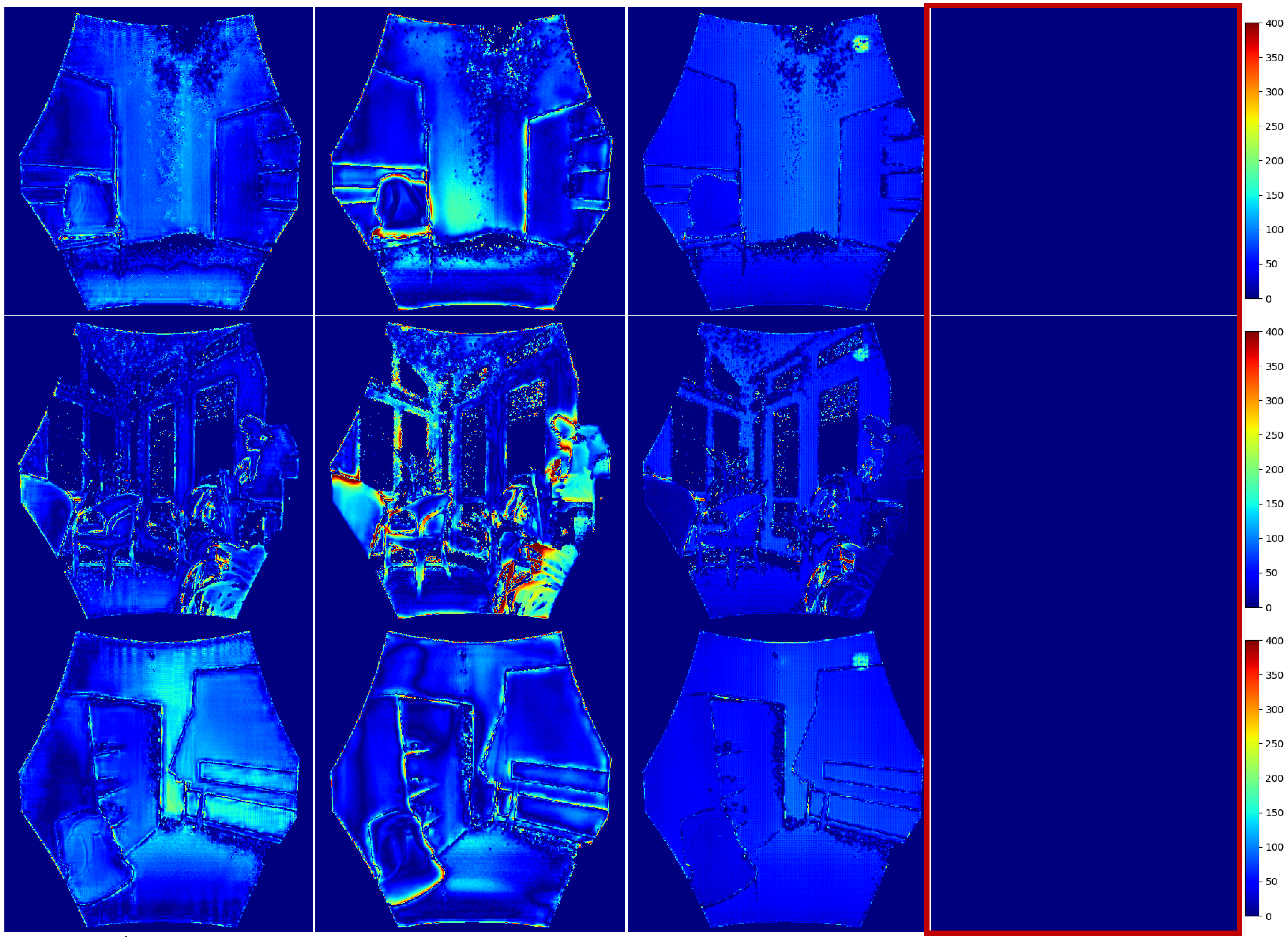



a) PConv b) Conv c) GAN d) LaMa

# Results - Qualitative Comparision Own dataset

Color-coded deltas of valid areas (less better)

LaMa no deltas



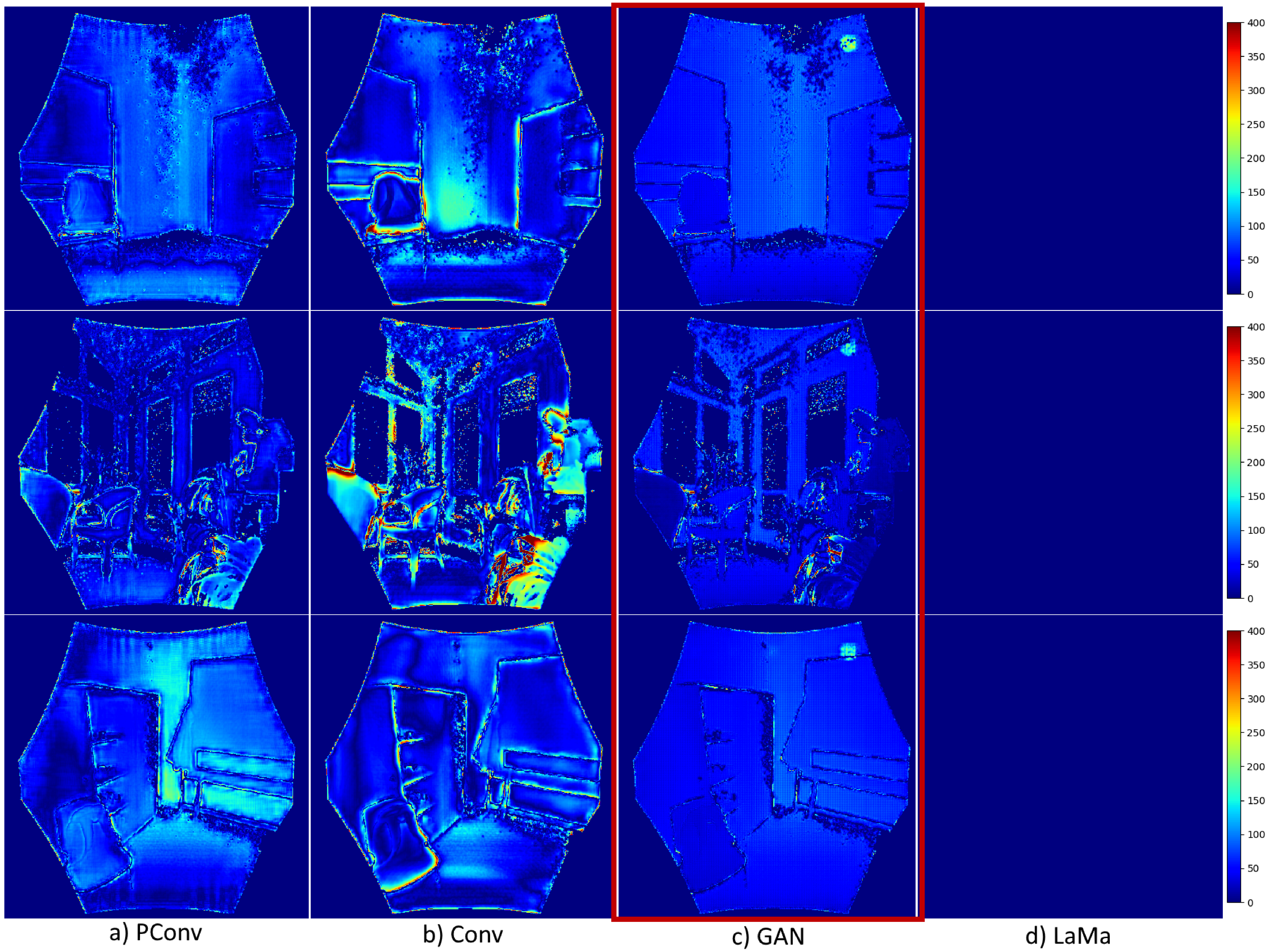

a) PConv b) Conv c) GAN d) LaMa

# Results - Qualitative Comparision Own dataset

Color-coded deltas of valid areas (less better)

LaMa no deltas

GAN only small (apart top right)



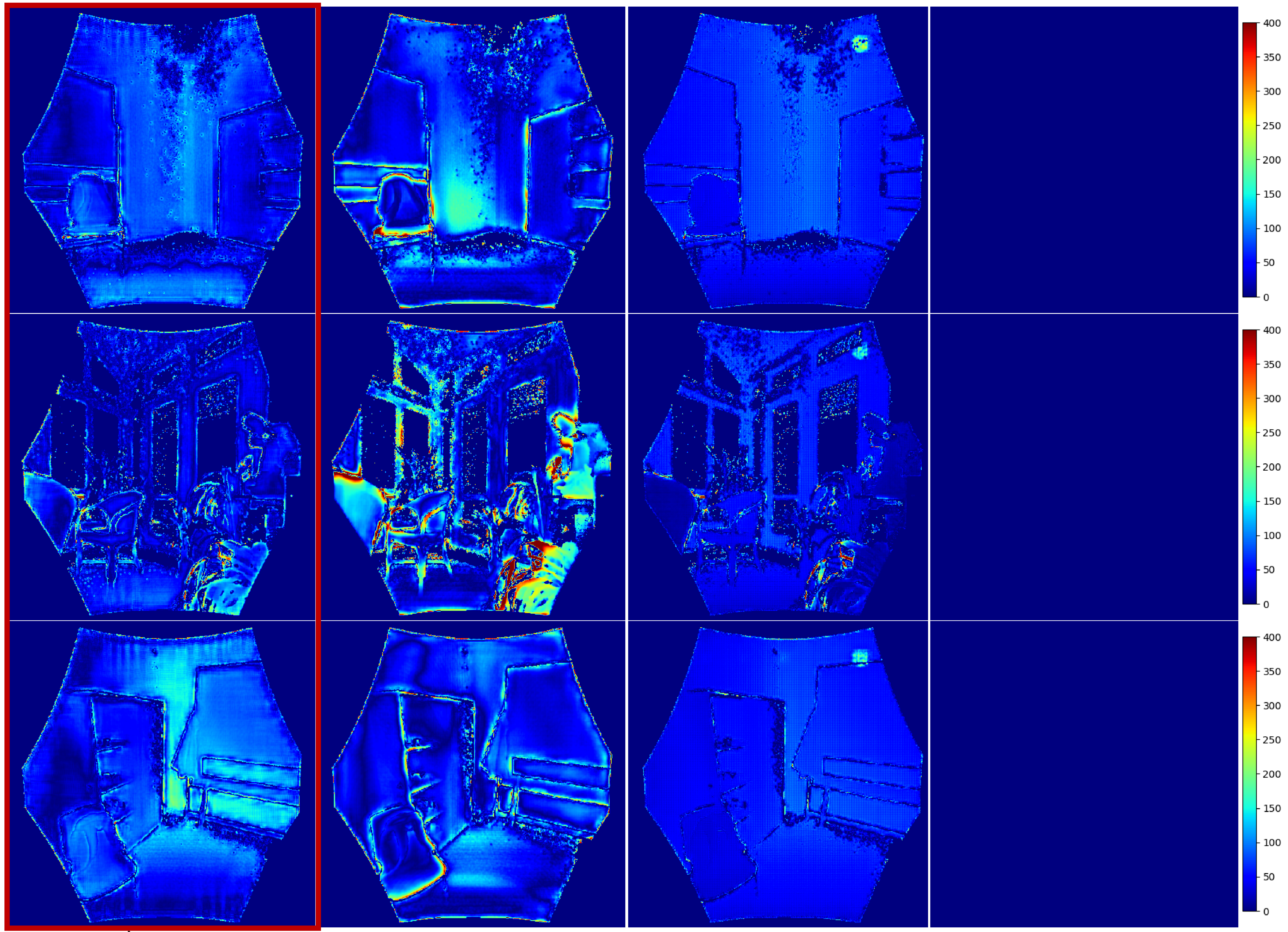
# Results - Qualitative Comparision Own dataset

Color-coded deltas of valid areas (less better)

LaMa no deltas

GAN only small (apart top right)

PConv medium




a) PConv b) Conv c) GAN d) LaMa

# Results - Qualitative Comparision Own dataset

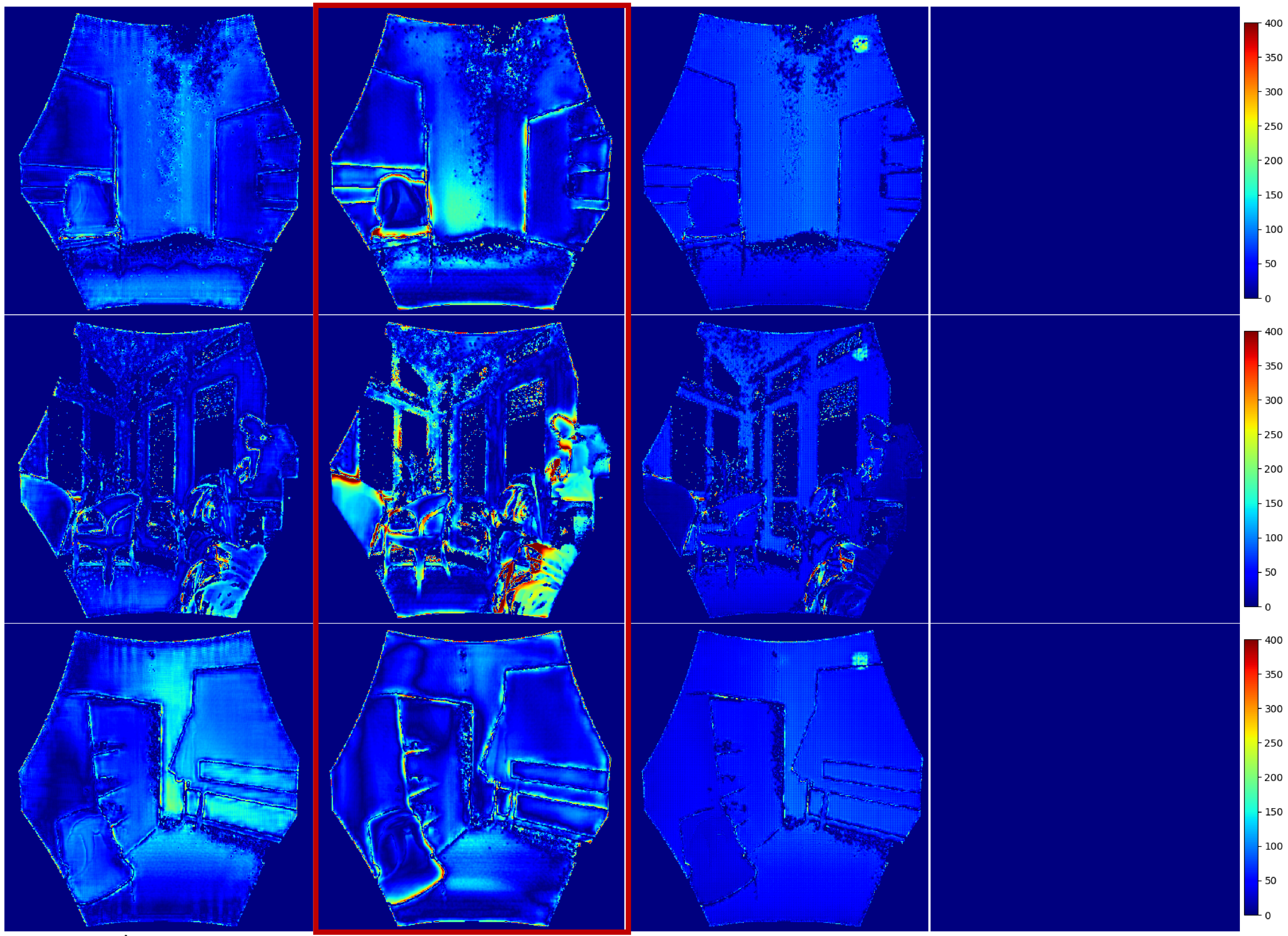
Color-coded deltas of valid areas (less better)

LaMa no deltas

GAN only small (apart top right)

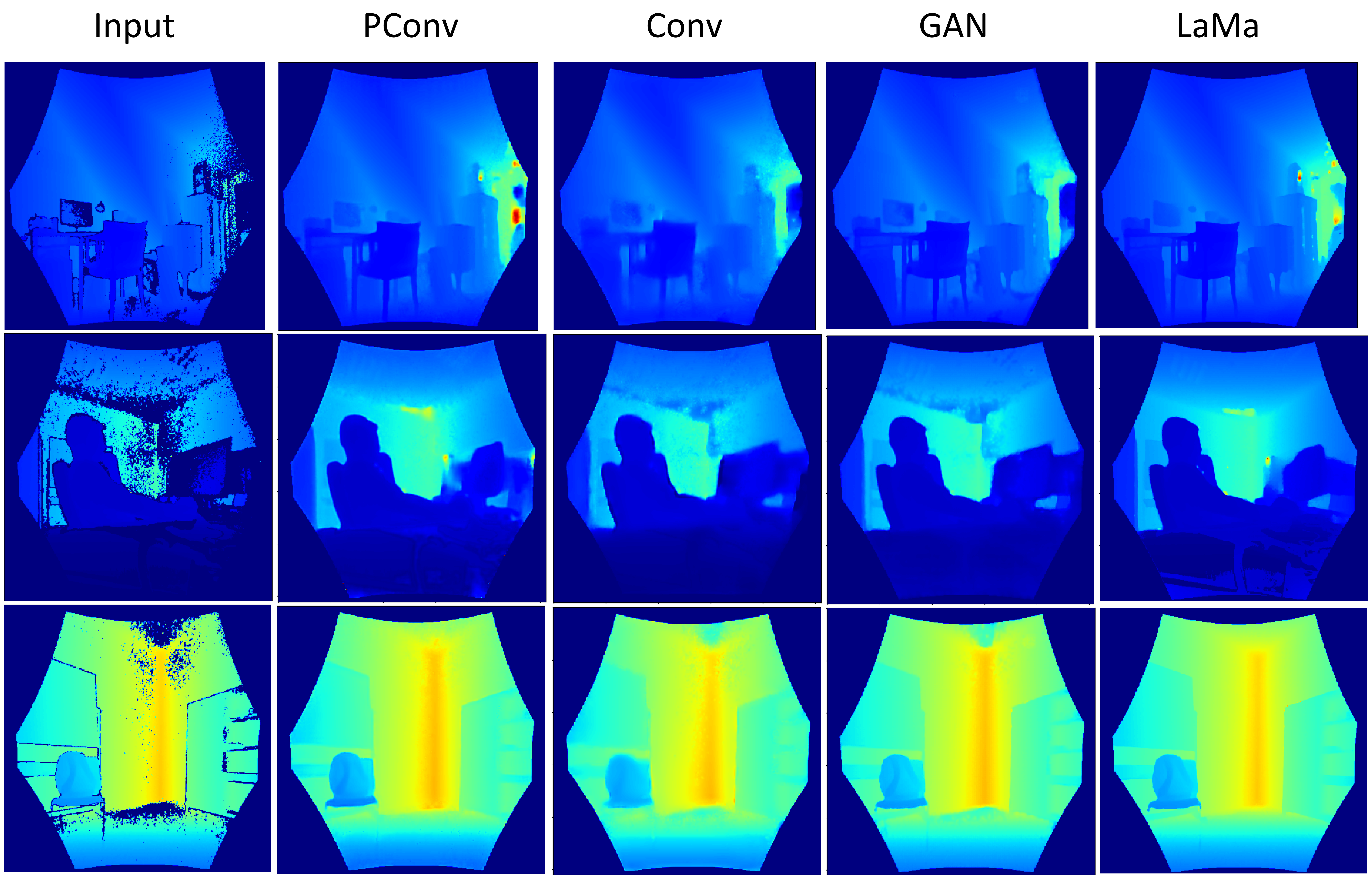
PConv medium

Conv the highest



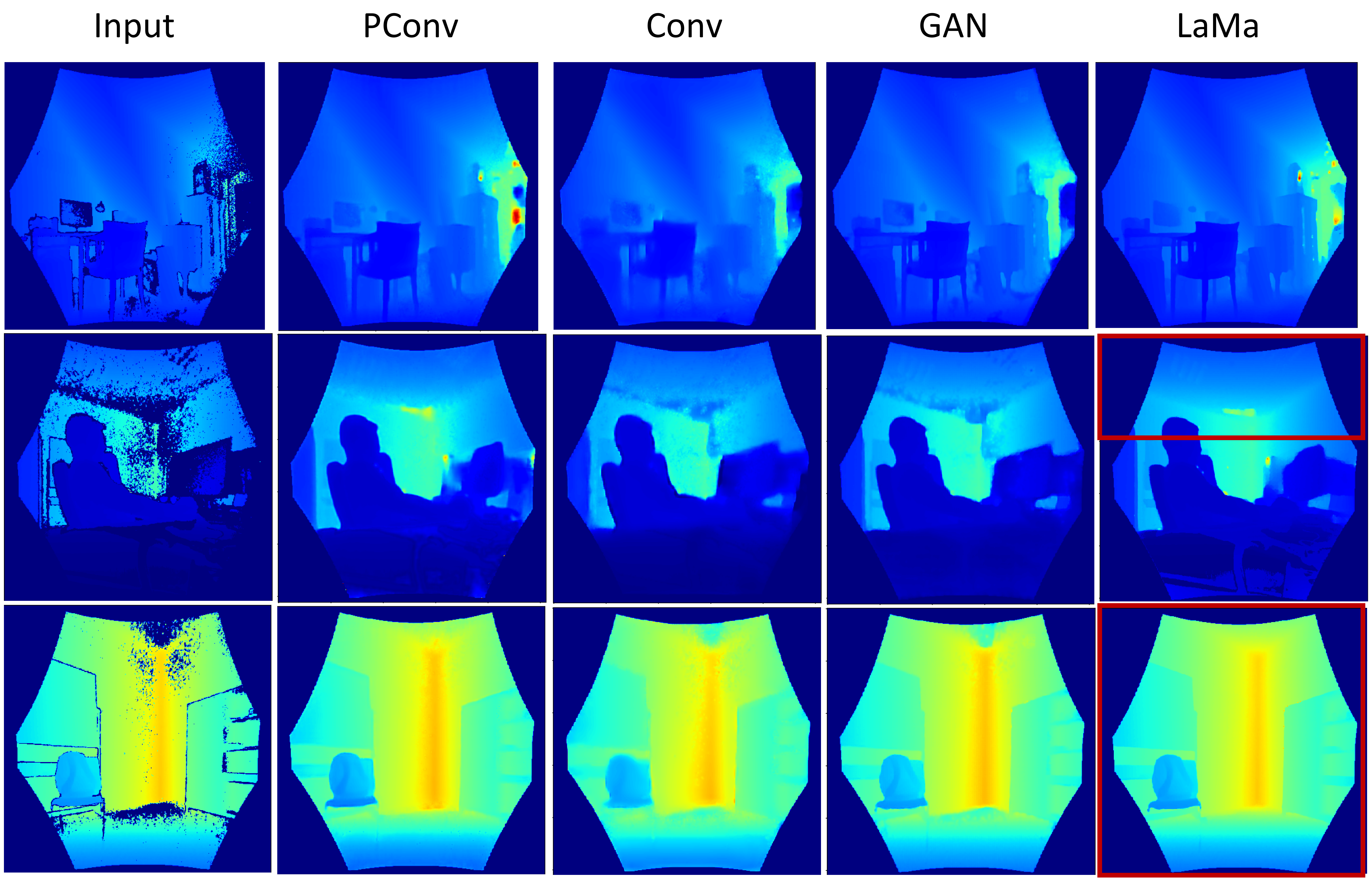

a) PConv b) Conv c) GAN d) LaMa

# Results - Qualitative Comparision Own dataset



# Results - Qualitative Comparision Own dataset

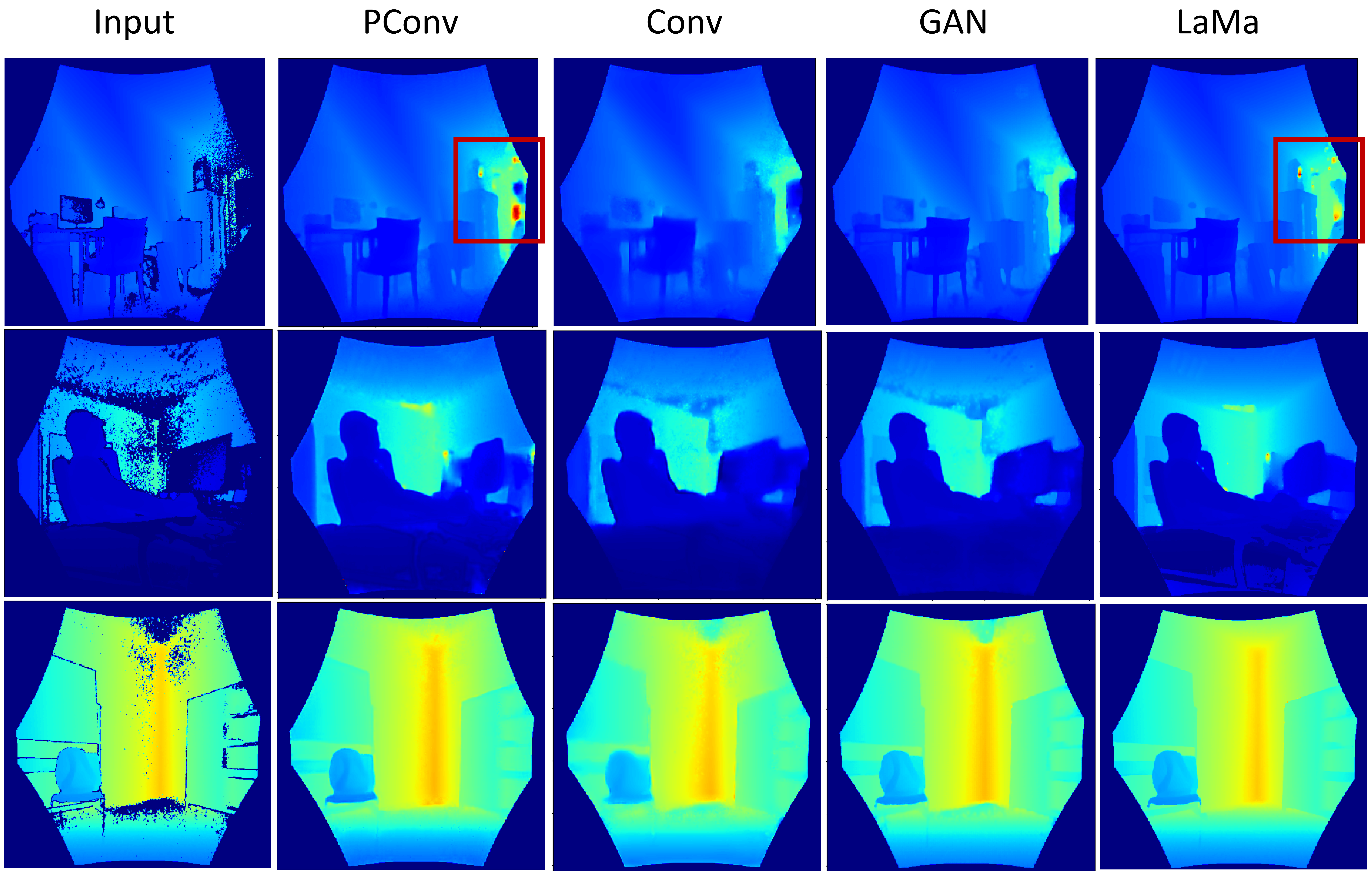
LaMa most often the best



# Results - Qualitative Comparision Own dataset

LaMa most often the best

PConv similar, both struggle with outliers

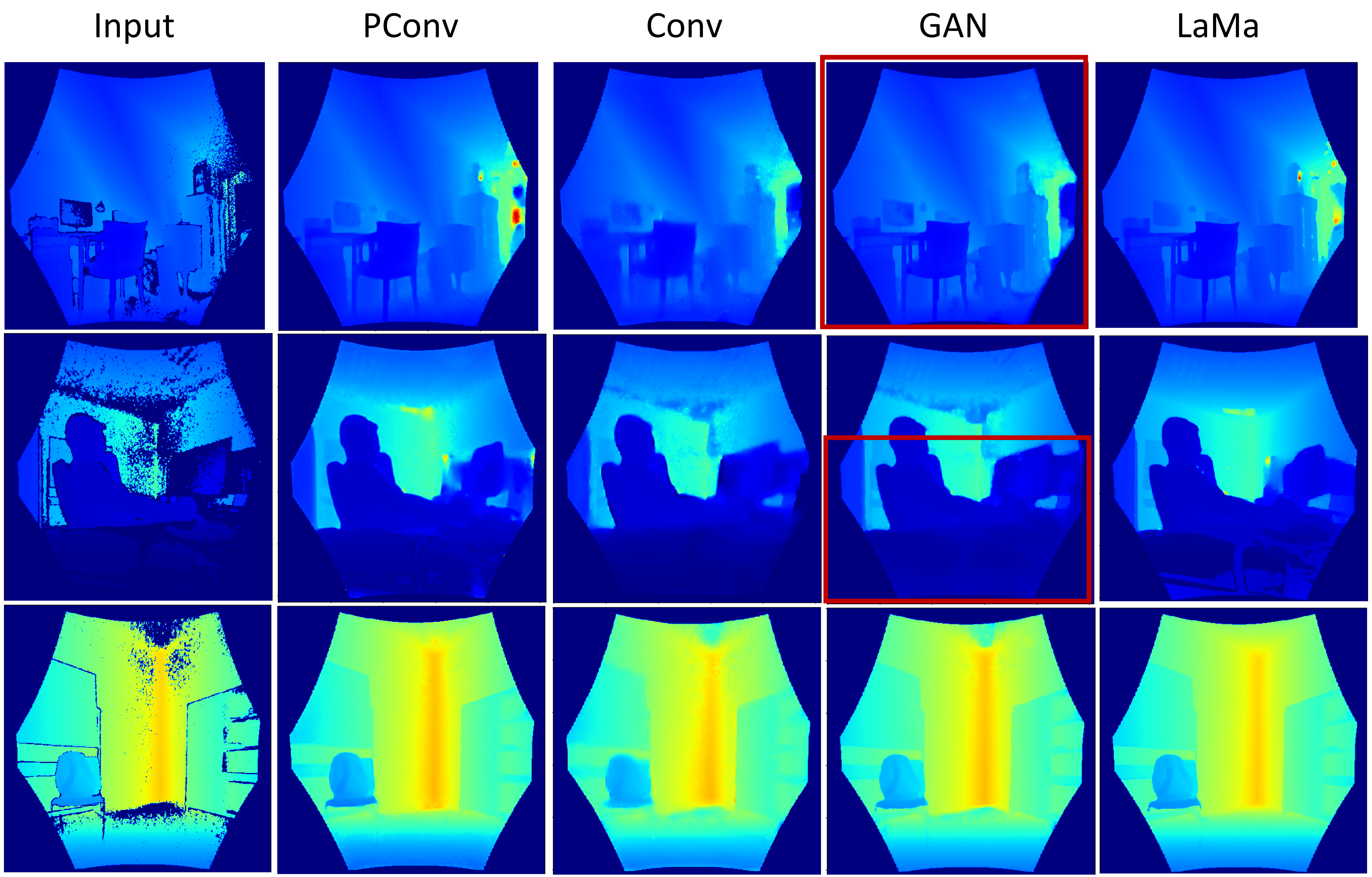


# Results - Qualitative Comparision Own dataset

LaMa most often the best

PConv similar, both struggle with outliers

Sometimes GAN the best



# Conclusion

- Investigated depth image inpainting using deep learning

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance
- Trained on NYUV2 with synthetic holes

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance
- Trained on NYUV2 with synthetic holes
- All models reasonably good

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance
- Trained on NYUV2 with synthetic holes
- All models reasonably good
  - LaMa best but slow (60ms)

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance
- Trained on NYUV2 with synthetic holes
- All models reasonably good
  - LaMa best but slow (60ms)
  - Part. Conv. U-Net, GAN (small holes) good, real-time-capable

# Conclusion

- Investigated depth image inpainting using deep learning
  - Real-time application
  - Without color guidance
- Trained on NYUV2 with synthetic holes
- All models reasonably good
  - LaMa best but slow (60ms)
  - Part. Conv. U-Net, GAN (small holes) good, real-time-capable
  - Highly scene-dependent

# Future Work

- Incorporate RGB data as optional input

# Future Work

- Incorporate RGB data as optional input
- Investigate transformer models (real-time) (use temporal coherency)

# Future Work

- Incorporate RGB data as optional input
- Investigate transformer models (real-time) (use temporal coherency)
- Produce ground truth for Azure Kinect (couple with stereo cam?)

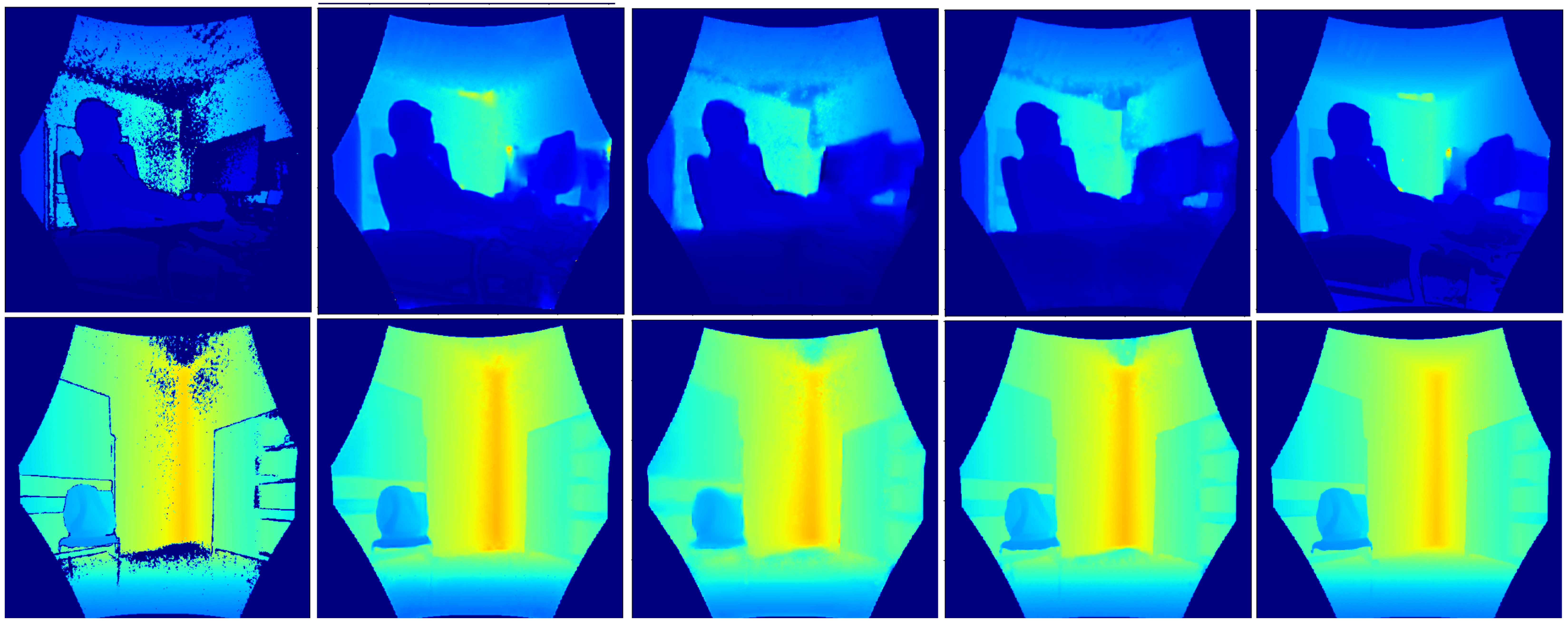
# Future Work

- Incorporate RGB data as optional input
- Investigate transformer models (real-time) (use temporal coherency)
- Produce ground truth for Azure Kinect (couple with stereo cam?)
- Produce accurate error model for Azure Kinect

# Future Work

- Incorporate RGB data as optional input
- Investigate transformer models (real-time) (use temporal coherency)
- Produce ground truth for Azure Kinect (couple with stereo cam?)
- Produce accurate error model for Azure Kinect
- Automatically switch model based on scene/holes

# Thank you for your attention!
# Questions?



r.fischer@uni-bremen.de