# Clicking Better Images with Under Display Cameras (UDC) in Smartphones

Umar Masud and Faraz Ali

**Abstract**—Under Display Cameras (UDC) are the next-generation technology developed for mobile devices where the camera is embedded under the display so that the screen covers the entire surface. However, placing the camera inside the screen introduces degradation such as noise, flare, haze and low light. In this work, we solve this ill-posed inverse problem and restore the UDC image through approaches of cross-model knowledge distillation and denoising diffusion probabilistic models. Distillation allows us to develop an efficient solution which is deployable, meanwhile, the use of diffusion models previously unexplored on UDC gives some interesting results, beating the state-of-the-art PSNR value.

**Index Terms**—Under Display Cameras, Denoising, Cross-Model Knowledge Distillation, Diffusion

## 1 Introduction

Recent advances in product technology have led to a new imaging system that places the camera in a device (phone, tablet, etc.) beneath the screen to provide the user with a bezel-less, full-screen experience. UDC replaces the current top-notch or punch-hole based cameras that break the screen's smoothness, improving the display-to-body ratio and enhancing eye contact, especially in video applications. However, placing the camera behind the screen brings inevitable degradations that cannot be neglected. Noise, flare, haze and low light are common degradation found in UDC images. There are two types of device screens - Transparent OLED (T-OLED) and phone Pentile OLED (P-OLED) which produce different UDC images. T-OLED produces a simpler degradation that has slight noise and blur without introducing any light or haze effects. This is because T-OLED is transparent, allowing most photons to pass through. On the other hand, P-OLED produces intense degradation like noise, haze, low light, etc. due to a lower light transmission rate. Consequently, it is much harder to restore a P-OLED output. An example of UDC image degradation can be seen in Fig 1.


Fig. 1. Sample of a noisy image obtained through T-OLED and P-OLED screens.

Deep Learning has been extensively used for solving such inverse problems due to the end-to-end learning capability of such models. They do not need any explicit information about the data distribution as long as there is some form of supervision. From convolution-based [1], [2] to transformer models [3], [4], there has been a plethora of research to solve such inverse problems. Specifically, UDC image restoration requires the joint modelling of methods resolving different optical effects caused by the displays and camera lens. Despite being a developing problem, various methods have been proposed to recover UDC Images owing to the work of [5].

However, two areas have still not received much attention for UDC restoration. Firstly, little thought has been given to developing efficient models that can be deployed successfully. Since we eventually need the solution to work on mobile devices, it makes sense to make a lightweight model. Secondly, the class of diffusion models [6] that are based on learning the noisy distribution of data and removing it at subsequent steps has not been explored at all. Thus, in our work, we experiment with two approaches - a) We perform cross-model knowledge distillation to get a lightweight, efficient solution that offers high performance at lesser computational cost, and b) We experiment with a pre-trained diffusion U-Net model to study the efficacy of denoising diffusion probabilistic methods. An 8-layer Denoising CNN (DnCNN) [2] estimates noise variance for inputs, serving as priors to the model. To our knowledge, we are the first to experiment with knowledge distillation and diffusion-based models for restoring UDC images.

Thus, our contributions can be summarized as -

- Considering efficiency and deployment, we implement a cross-model knowledge distillation between a simple convolution-based U-Net student and a bulky, pre-trained Transformer teacher.
- We evaluate the capability of denoising diffusion probabilistic models (DDPM) on UDC image restoration by fine-tuning a pre-trained diffusion U-Net model in conjunction with an auxiliary DnCNN for noise variance estimation.

## 2 Related Work

Zhou et al. [5] first introduced the problem of UDC image restoration by providing an analysis of the optical systems underlying 4k Transparent OLED (T-OLED) and phone Pentile OLED (P-OLED) imaging setups. The authors first compared the two degradation types through their display pattern, corresponding point spread function (PSF) and light

transmission rate. Their findings demonstrated the stripe-like pattern of T-OLED displays leading to a horizontal spread of light in the PSF while P-OLED displays exhibited a pentile layout similar to the structure of an RGBG matrix, resulting in an even distribution of light. Moreover, they noted that a lower transmission rate in P-OLED displays was attributed to higher scattering of photons due to finer pixel layout and higher photon absorption in the polyamid substrate used for the screen. The authors also presented a Monitor-Camera Imaging System (MCIS) which was composed of a point-grey camera focused on a 4K LCD display that was projecting a given image. In front of the camera lens was either a T-OLED, P-OLED, or glass (no display) panel. This enabled the authors to capture paired data of high-quality and degraded images while measuring intensity scaling factor, read noise, shot noise, and other imaging parameters. The parameters were subsequently used for generating synthetic degraded data. Finally, they presented baseline performance metrics with a Wiener Filter restoration pipeline and deep learning-based pipelines with a U-Net model achieving the highest PSNR of 36.71 and 30.45 for T-OLED and P-OLED respectively.

In 2020, Zhou et al [7] held a competition at the European Conference on Computer Vision (ECCV) for UDC image restoration. The competition employed the same MCIS system from [5] to share a dataset of 300 paired images from the DIV2K dataset with the participants. The Baidu Research Vision team performed the best for T-OLED restoration by utilizing a dense residual network for image denoising and demosaicking. The team added a shade-correction module which learns coefficients for patterns specific to the T-OLED screen to correct shade in addition to normal restoration. By training the model on a patch size of 128x128, the team scored a 38.23 PSNR and an SSIM of 0.9803 for T-OLED. Moreover, the CET_CVLab achieved the best performance for P-OLED restoration through a Pyramidal Dilated Convolutional RestoreNet (PDCRN). Their model followed an encoder-decoder architecture, with the downsampling in the encoder occurring via a discrete wavelet transform (DWT) and an inverse discrete wavelet transform (IDWT) upsampling the data in the decoder. Dilated convolutional pyramids that stack dilated convolution layers with decreasing dilation rates are also interspersed throughout the model, to help minimize information loss. The team achieved a PSNR of 32.99 and an SSIM of 0.9578 for P-OLED.

Another competition was held in ECCV 2022 featuring a UDC restoration track [8]. The dataset used was inspired by work done in [9] that stemmed from the original MCIS work but instead reformulated the problem to also account for diffraction flare in saturated regions of the high-dynamic range (HDR) image. The top result was achieved by the USTC_WXYZ team which employed a multi-input multi-output deep convolutional neural network with various dense residual blocks, attention modules, and cross-fating fusion modules for multi-scale feature fusion. The model resulted in the highest competition PSNR of 48.48 with an SSIM of 0.9934.

In terms of methods with efficient deployment, Conde et al [10] achieved competitive results for UDC image restoration using only deep learning methods while having 4x less compute operations. The authors first develop DRM-UDCNet, a CNN with an encoder and decoder architecture, where each segment contains several Dense Residual Modules (DRM). Additionally, a parallel attention branch performs channel and spatial attention on the input image separately before combining the extracted features with the base branch. The authors then optimize the architecture into a more efficient model, termed the LUDCNet, which contains fewer DRM blocks, no batch normalization, and processes images at half-resolution while upsampling them at the end. The authors mainly train and evaluate on the SYNTH dataset used in [8]. The larger DRM-UDCNet achieves a 40.21 PSNR and 0.98 SSIM on the dataset with 2.9 million parameters, while the LUDCNet achieves a 0.93 SSIM with around 300K parameters.

## 3 Proposed Method

### 3.1 U-Net with Knowledge Distillation

In this approach, we attempt to develop a lightweight model that performs well given the limited training data using the technique of Knowledge Distillation (KD). KD is an approach to transfer the representation capacity of a larger model (teacher) to a smaller model (student) that can be practically deployed under real-world constraints, without significant loss in performance. [11] showed the efficacy of such a technique. In our case, we specifically perform cross-model knowledge distillation, where we distil feature-level knowledge from a transformer model to our simpler U-Net model. We first train a transformer model - Restormer [4] on our training pair of data, achieving very good results. However, it is not practical to deploy this model and inference in real-time on mobile devices, so we take another simpler architecture U-Net [12] and train it on the available data. The choice for U-Net was made based on its encoding-decoding property and useful skip connections.

Since the training data is limited (only 240 images) and the learning capacity of the U-Net base is not as strong, we direct it to learn and output better representations by using the high-performing Restormer model. During training, we freeze the teacher model so it is only working in evaluate mode while the U-Net is allowed to backpropagate and update its weights.

The challenge with feature-level distillation between different architectures (cross-model) is that each feature map at a particular layer of the two models does not need to have the same shape or learn the same features. To overcome this, there needs to be a reduction mapping before we can transfer the knowledge from teacher to student. We experimented with two such reduction operators and found the following:

1) 1x1 convolution - First we tried to combine the feature maps by maintaining their spatial output and compressing the information along the channel through the use of 1x1 convolution. In particular, we perform the following combination -

$$\text{Student Layer (B, C}^{'}, H, W) \leftarrow TeacherLayer(B, C^{"}, H, W)$$

The 1x1 convolution converts the $C^{"}$ channels to 1 i.e. single channel and the result is then

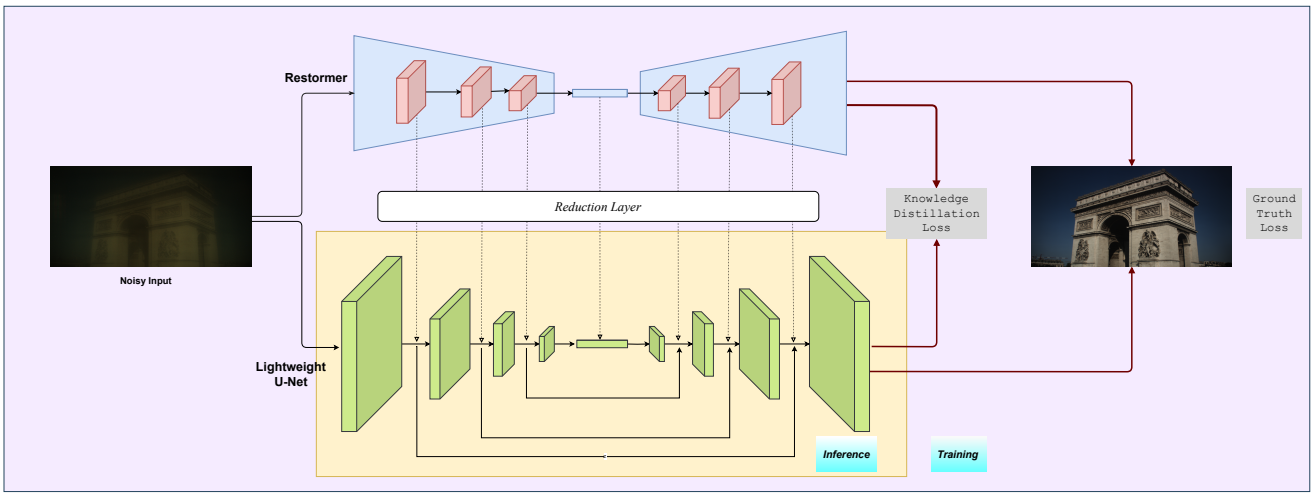


Fig. 2. The teacher model is Restormer [4] which is pre-trained and frozen, the lighter student model is a U-Net that does 4x downsampling and upsampling.

simply added (or multiplied). However, it was found that since convolution is *parametric*, it was absorbing all the information that should have been transferred to the student model layers. As a result during inference, when the 1x1 convolutions were removed, the student model lost all its ability to generate results.

2) Global Average Pooling - In this, we simply squeeze the information along the spatial output. We then take the mean along the channel axis as well, returning us a single scalar value compressing all the information of the feature map, which is then added/multiplied with the U-Net layer.

$$\text{GAP(B, C}^{"}\text{}, H, W) \rightarrow (B, C^{"}, 1, 1).mean(1) \rightarrow (B, 1, 1, 1)$$

Being *non-parametric*, this type of pooling forced the student model's layers to adjust the weights and learn accordingly, working well even during inference when there is no extra knowledge.

The architecture of our models and training with knowledge distillation can be visualised in Fig. 2.

### 3.2 Diffusion U-Net

A popular approach to image generation and restoration tasks has been denoising diffusion probabilistic models (DDPM) [6]. The diffusion approach is split up into two stages: a forward diffusion process which gradually adds Gaussian noise to an input image, and a reverse diffusion process which samples a noisy image from a Gaussian distribution and then incrementally removes noise until a new image is uncovered [6]. Starting with an input image, $x_0$, over $T$ timesteps, the forward process gradually adds Gaussian noise with the Gaussian noise added at timestep $t < T$ having a mean, $\mu_t = \sqrt{1-\beta_t}\mathbf{x}_{t-1}$, and variance, $\beta_t$ for T timesteps. The variances used across timesteps are scheduled such that the final image after $T$ timesteps is nearly pure Gaussian noise. In the reverse diffusion process, a neural network is employed to learn the mean ($\mu_\theta$) and variance ($\sum_\theta$) parameters of the conditional probability distribution to recover the incrementally denoised example at each timestep:

$$p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mathbf{x}_{t-1}; \mu_\theta(\mathbf{x}_t, t), \sum_\theta(\mathbf{x}_t, t))$$

Through re-parameterization tricks, the objective function for this reverse process can be made into a simpler L2 loss minimization between the Gaussian noise distribution sampled during the forward process at time $t$, $\epsilon$, and the model's approximation of the noise, $\epsilon_\theta(\mathbf{x}_t, t)$, parameterized by the input noisy image, $\mathbf{x_t}$ and timestep, $t$ [6].

For this work, we adopt and fine-tune the standard diffusion-based U-Net model introduced in [13]. The model is initially pre-trained on the Flickr-Faces-HQ (FFHQ) dataset which consists of roughly 70K PNG images. The model itself is a modified version of a standard U-Net architecture, with 6x downsampling and 6x upsampling layers. Each layer is composed of a single residual CNN block with sinusoidal time-step embeddings added to it. At the 16x16 resolution, there is a self-attention layer used following the layer's residual block. There is also a larger variant with more down-/up-sampling and attention layers which was pre-trained on the larger ImageNet128 dataset [14].

At each forward pass, the model requires both the input noisy image and the timestep corresponding to the point in time in the forward diffusion process when the input image's Gaussian noise was sampled. Since we aim to only fine-tune the model, we forgo the forward diffusion process and train the reverse diffusion process on our dataset. As a result, the timestep is unknown during each forward pass through the model. To address this, we create a simple 8-layer DnCNN which takes the noisy input image and outputs a scalar estimate of the noise variance in the input. This scalar estimate can be used through the pre-computed variances that the diffusion U-Net model was trained with to find the timestep of the closest known variance and thus, input that timestep along with the noisy input image to the diffusion U-Net. The model architecture is shown in Fig 3

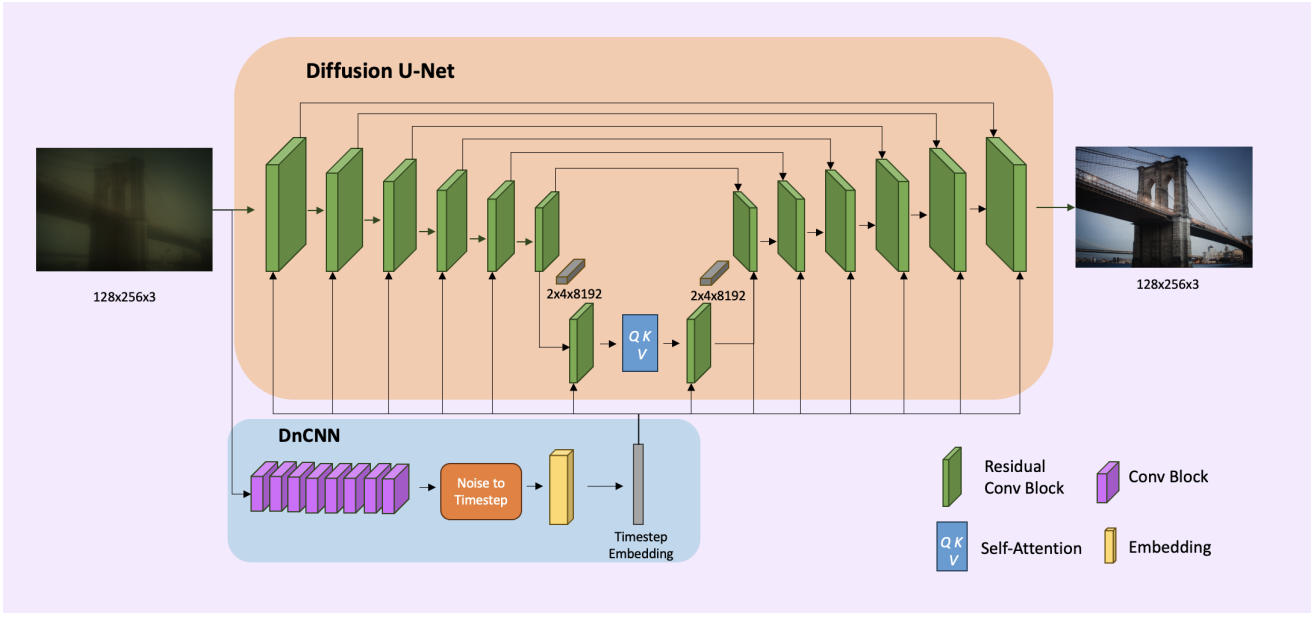


Fig. 3. Diffusion U-Net model with auxiliary DnCNN for noise variance estimation priors.

The DnCNN architecture typically outputs a feature map of the estimated noise, $\mathbf{z} \in \mathbb{R}^{BxCxHxW}$. To convert this into a scalar noise variance estimate, $\mathbf{v} \in \mathbb{R}^{Bx1}$, instead, two approaches were tried:

1) Double Average Pooling - The simplest approach was to perform a 2D Global Average Pooling operation over the spatial dimensions of each output channel, followed by another Average Pooling operation over the remaining channel dimension:

$$\mathbf{v} = AP(GAP(\mathbf{z}))$$

During experimentation, it was found that this approach did not work as well in approximating the noise variance in the input image. The DnCNN was unable to produce feature maps in a way that would result in averaging the features across all three dimensions producing a strong estimate of the variance in the noise predicted.

2) Dense Layer - The second approach involved performing only one 2D Global Average Pooling operation over the spatial dimensions of the output feature map, and then passing the result to a dense layer with only 1 output neuron:

$$\mathbf{v} = GAP(\mathbf{z})\mathbf{w} + b, \mathbf{w} \in \mathbb{R}^{Cx1}$$

This worked better in practice for estimating noise variance. It could perhaps be explained by the presence of the extra set of weights in the final dense layer potentially permitting more capacity for the DnCNN to model the residual error present in its initial feature map estimation of the noise.

## 4 Experimental Results

### 4.1 Dataset

We use the dataset provided by [5] which was also used in the ECCV 2020 competition [7]. The dataset comprises 240 pairs of clean/noisy images, for both the T-OLED and P-OLED types. We use only this set of images, without any additional training.

### 4.2 Training Parameters

The knowledge distillation models were trained for 200 epochs while the diffusion U-Net models were trained for 100 epochs. All experiments were done with a learning rate of $1e^{-}3$ using the Adam optimizer. Due to computational constraints, we trained with a resized input of (256, 512, 3) with a batch size of 4. Many loss functions are used in reconstruction tasks such as L1, L2, SSIM, Grad Loss, Perceptual Loss, etc. In this work, we use L2 and Perceptual loss (through VGG16) for knowledge distillation and ground truth comparison. For the diffusion models, we only use the L2 loss. Training took place on a single accelerator - either an NVIDIA RTX A4000 or an NVIDIA RTX 4090.

### 4.3 Quantitative Results

As seen in Tables 1 and 2, through knowledge distillation, we can train a model for UDC restoration with as low as 7.78M parameters, achieving competitive results in both categories of datasets. In P-OLED, the U-Net base (i.e. without distillation) performs well enough to get 27.72 PSNR compared to ground truth. With distillation, we achieved a PSNR of 30.59 with 0.91 SSIM. On T-OLED restoration, it is found that the U-Net base works a little better without distillation, getting 37.75 PSNR with 0.98 SSIM meanwhile with distillation it gets 36.24 PSNR with 0.97 SSIM.

The standard diffusion U-Net model outperformed all other methods on the T-OLED data, achieving a PSNR of

TABLE 1
Performance metrics comparison between different approaches on T-OLED data.

| Approach | PSNR (dB) | SSIM | No. of Parameters | Inference Time (s/img) | CPU/GPU |
|---|---|---|---|---|---|
| Best from [7] | 38.23 | 0.98 | – | 11.8 | Tesla M40 |
| U-Net Base | 37.75 | 0.98 | 7.78M | 0.03524 | RTX A4000 |
| U-Net Base + KD | 36.24 | 0.97 | 7.78M | 0.03588 | RTX A4000 |
| Diffusion U-Net (Standard) | **42.37** | **0.99** | 94M | 0.27768 | RTX 4090 |
| Diffusion U-Net (Large) | 30.33 | 0.9 | 553M | 0.86504 | RTX 4090 |

TABLE 2
Performance metrics comparison between different approaches on P-OLED data.

| Approach | PSNR (dB) | SSIM | No. of Parameters | Inference Time (s/img) | CPU/GPU |
|---|---|---|---|---|---|
| Best from [7] | **32.9** | **0.96** | – | 0.044 | Tesla T4 |
| U-Net Base | 27.72 | 0.91 | 7.78M | 0.03532 | RTX A4000 |
| U-Net Base + KD | 30.59 | 0.91 | 7.78M | 0.03616 | RTX A4000 |
| Diffusion U-Net (Standard) | 27.15 | 0.83 | 94M | 0.27724 | RTX 4090 |
| Diffusion U-Net (Large) | 18.09 | 0.47 | 553M | 0.86636 | RTX 4090 |

42.37 and a 0.99 SSIM. However, for P-OLED data, the standard diffusion U-Net achieved a lower PSNR at 27.15 and an SSIM of 0.83 as compared to the knowledge distillation approaches. Moreover, in both degradation types, the larger diffusion U-Net performed worse in both PSNR and SSIM than all other methods.

In terms of throughput, the inference times shown in Tables 1 and 2 are computed through performing evaluation on original size data (as memory constraints were an issue during training). It is evident that the U-Net only models tend to process input images the quickest, without accounting for differences in hardware. Since the eventual goal of any image restoration pipeline for smartphones is to be utilized in a resource-constrained, real-time setting, the results from Tables 1 and 2 can be further visualized to gauge which of the approaches is most viable. By plotting the SSIM scores against the ratio of the PSNR and inference times of each method, as seen in Fig. 5, the more viable approaches for real-time usage are found to be U-Net Base and U-Net Base + KD models amongst the novel approaches tried.

### 4.4 Qualitative Results

We plot the model outputs obtained through knowledge distillation and denoising diffusion in Figure 4. The results greatly resemble the original ground truth images, matching the contrast, clarity and sharpness, proving that our models could reverse the degradations quite well. It effectively restores the low-frequency components maintaining the minute details in reconstruction. However, we also notice that diffusion models tend to over-correct the colours and typically result in different shades of a given hue.

## 5 Discussion

Overall our methods lay a good groundwork for experimenting with new approaches. KD works in the case of P-OLED, improving the PSNR from 27.72 to 30.59, however, a similar trend is not seen in T-OLED where the base U-Net without distillation performs slightly better. There could be two possible reasons for that - a) T-OLED is a fairly simpler problem than P-OLED, where the nature of degradation is a lot like Gaussian blur and noise, so knowledge distillation might be overfitting for the problem and thus showing poorer results on test data, b) We perform cross-model knowledge transfer, where the teacher and student models are not identical, thus we do not know for certain that the information combined from the corresponding layers between the two models might be learning the same type of features.

The standard diffusion model performed the best out of all other approaches in restoring the T-OLED degraded images, with the highest PSNR of 42.37. This was likely because the diffusion U-Net model was first pre-trained to effectively remove Gaussian noise from sampled images and produce high-quality face images. As the task of restoring degradations on T-OLED images consists of mainly addressing some noise and blur, it is very close to the original pre-training task of Gaussian denoising. As a result, the pre-trained diffusion U-Net was able to easily transfer its learning from the initial task to denoising the UDC images in the T-OLED setting. Comparatively, the diffusion approaches did not perform as well on the P-OLED data. This could be explained by the fact that the P-OLED screens introduce an amalgamation of different degradations, such as haze, low light, blur, and flare, which results in a much more complex restoration task. As this task is functionally more challenging than the initial pre-training task of removing Gaussian noise, it is expected that the diffusion models will not perform as well just after fine-tuning.

In terms of qualitative results, the diffusion models exhibited an interesting artefact of over-correcting the colour intensity in the P-OLED restored images. This might have been due to the original training data (FFHQ) not containing similar natural scene images (ex. facial images with similar background colours) as the natural scene images in our dataset. Furthermore, since our dataset was only 240

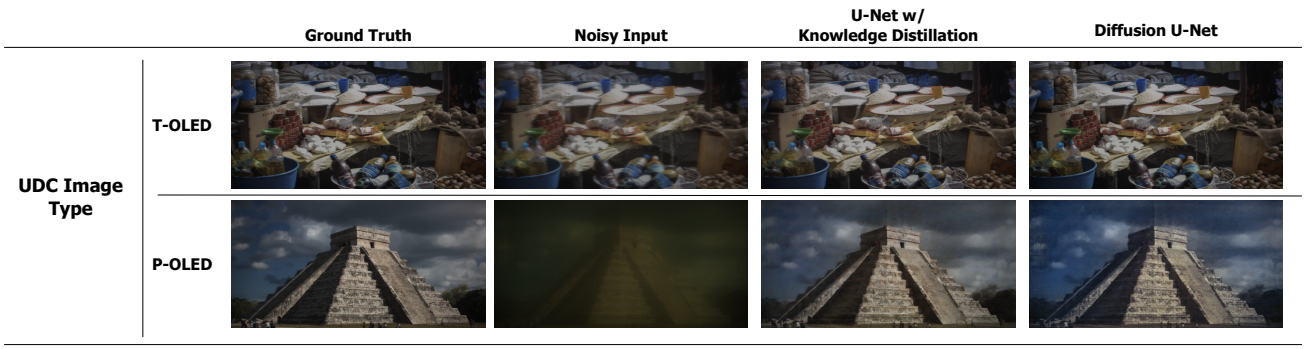


Fig. 4. Comparison of sample outputs for T-OLED and P-OLED degradations between approaches.

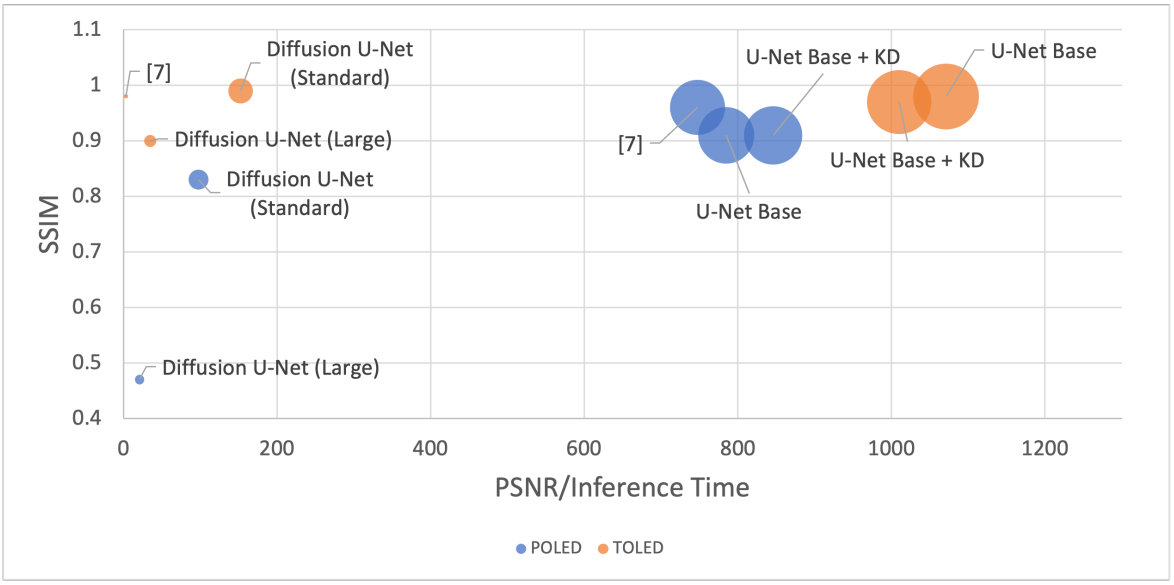


Fig. 5. SSIM vs. PSNR/Inference Time for various approaches. Scale of each circle is the product of its values on each axis.

training examples, this would not have been sufficient to effectively learn accurate colour representations for larger models like DDPMs. Similarly, the larger diffusion U-Net model's poor performance in both T-OLED and P-OLED settings can be attributed to the lack of training data volume. Since the model had around 500M parameters, using a small training dataset would likely not be enough to effectively shift the parameters of the model without overfitting the training data.

Finally, we also tried a few other experiments that did not give satisfactory results. In particular, we tried to do some preprocessing on the inputs that could effectively make it easier for a simple DnCNN-only model to restore the images. For this, we tried re-lighting through the Retinex algorithm [15] (centre-surround, path-based, and DeepRetinex [16]) and dehazing via dark channel prior [17]. Both of these preprocessing failed to give better results, decreasing the quality by up to 5dB PSNR.

## 6 Conclusion

In this work, we dealt with restoring under display camera images through knowledge distillation and diffusion approaches. We developed a lightweight, efficient model through knowledge distillation between a larger transformer-based model and a more efficient U-Net base - demonstrating the effectiveness of distillation by improving the results. The knowledge distilled U-Net base model provided the quickest inference times than other methods and achieved decent PSNR and SSIM metrics on both T-OLED and P-OLED data. We also experimented with diffusion models that are so far not yet explored for UDC restoration and observed that they might work quite well, beating state-of-the-art results as well on simpler degradations (eg. images from T-OLED screens) with just fine-tuning.

In future work, we could experiment with making the models more efficient via quantization and pruning. The cross-model knowledge distillation connections can also be further explored through ablation studies to find the most optimal connections between feature maps of different layers. For the diffusion models, as they are too computationally and memory-expensive to deploy in real-time, a knowledge distillation approach could also be experimented with as well. Larger datasets, like [8], should also be experimented with, and performance in real-time settings (eg. on an accelerated smartphone device) may also be explored. There should also be more endeavour to get better and larger datasets. Also, treating UDC as a non-blind

image formation could help develop deconvolution-based solutions (through PSF estimation) that can be embedded right into the ISP level.

## Acknowledgments

The authors would like to thank Prof. David Lindell for his guidance throughout the project.

## References

[1] Q. Xu, C. Zhang, and L. Zhang, "Denoising convolutional neural network," in *2015 IEEE International Conference on Information and Automation*, 2015, pp. 1184–1187.

[2] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, p. 3142–3155, Jul. 2017. [Online]. Available: http://dx.doi.org/10.1109/TIP.2017.2662206

[3] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin transformer: Hierarchical vision transformer using shifted windows," 2021.

[4] S. W. Zamir, A. Arora, S. Khan, M. Hayat, F. S. Khan, and M.-H. Yang, "Restormer: Efficient transformer for high-resolution image restoration," 2022.

[5] Y. Zhou, D. Ren, N. Emerton, S. Lim, and T. Large, "Image restoration for under-display camera," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 9175–9184.

[6] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," 2020.

[7] Y. Zhou, M. Kwan, K. Tolentino, N. Emerton, S. Lim, T. Large, L. Fu, Z. Pan, B. Li, Q. Yang, Y. Liu, J. Tang, T. Ku, S. Ma, B. Hu, J. Wang, D. Puthussery, H. P. S, M. Kuriakose, J. C. V. au2, V. Sundar, S. Hegde, D. Kothandaraman, K. Mitra, A. Jassal, N. A. Shah, S. Nathan, N. A. E. Rahel, D. Chen, S. Nie, S. Yin, C. Ma, H. Wang, T. Zhao, S. Zhao, J. Rego, H. Chen, S. Li, Z. Hu, K. W. Lau, L.-M. Po, D. Yu, Y. A. U. Rehman, Y. Li, and L. Xing, "Udc 2020 challenge on image restoration of under-display camera: Methods and results," 2020.

[8] R. Feng, C. Li, S. Zhou, W. Sun, Q. Zhu, J. Jiang, Q. Yang, C. C. Loy, J. Gu, Y. Zhu *et al.*, "Mipi 2022 challenge on under-display camera image restoration: Methods and results," in *European Conference on Computer Vision*. Springer, 2022, pp. 60–77.

[9] R. Feng, C. Li, H. Chen, S. Li, C. C. Loy, and J. Gu, "Removing diffraction image artifacts in under-display camera via dynamic skip connection network," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 662–671.

[10] M. V. Conde, F. Vasluianu, S. Nathan, and R. Timofte, "Real-time under-display cameras image restoration and hdr on mobile devices," in *European Conference on Computer Vision*. Springer, 2022, pp. 747–762.

[11] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," 2015.

[12] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," 2015.

[13] J. Choi, S. Kim, Y. Jeong, Y. Gwon, and S. Yoon, "Ilvr: Conditioning method for denoising diffusion probabilistic models. in 2021 ieee," in *CVF international conference on computer vision (ICCV)*, 2021, pp. 14 347–14 356.

[14] P. Dhariwal and A. Nichol, "Diffusion models beat gans on image synthesis," *Advances in neural information processing systems*, vol. 34, pp. 8780–8794, 2021.

[15] E. H. Land and J. J. McCann, "Lightness and retinex theory." *Journal of the Optical Society of America*, vol. 61 1, pp. 1–11, 1971. [Online]. Available: https://api.semanticscholar.org/CorpusID:14430259

[16] C. Wei, W. Wang, W. Yang, and J. Liu, "Deep retinex decomposition for low-light enhancement," 2018.

[17] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.