Anomaly-Preserving Image Restoration from Noisy and Blurred Images using RCAN-like model

Riya Mahesh (EE21B112), Satvik Malapaka (EE21B080)
Vaibhav Krishna Garimella (EP21B040), Sanket Singh (EE21B118)
Department of Electrical Engineering, IIT Madras

Abstract— **Preserving anomaly visibility while removing noise and blur is essential in quality control, medical imaging, and automated anomaly detection. Conventional denoising often erases subtle but crucial anomaly details, which impacts diagnostic reliability and machine learning model accuracy. This study focuses on enhancing image clarity while preserving anomaly structures, ensuring that automated systems and analysts have high-quality visuals without compromising key diagnostic information for reliable assessments. The MVTec Anomaly Detection (AD) dataset has been used for training and validation.**

I. INTRODUCTION

In recent years, the task of image denoising has gained significant attention, particularly in applications where anomaly visibility is critical. This work aims to develop techniques for denoising images that contain anomalies, ensuring that these important features are distinguished from noise and preserved throughout the process. To achieve this, we implemented two primary architectures: a basic AutoEncoder-based model named UNet, and a more complex ResNet-like model that incorporates convolutional layers alongside residual attention blocks, known as Residual Channel Attention Network (RCAN). We rigorously evaluated the performance of both models using standard metrics such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM). Furthermore, we plotted object-wise performance to compare the effectiveness of each architecture in preserving anomaly details while enhancing overall image quality.

II. UNET ARCHITECTURE

As a first attempt, we implemented a simple U-Net based achitecture [2] consisting of convolutional encoder and decoder layers. The model comprises of the following layers:

- **Conv2d(3, 128, kernel_size=3, padding=1)**: Applies a convolutional layer with 128 filters and ReLU activation.
- **ReLU Activation** - Applies a ReLU activation function for non-linearity.
- **Conv2d(128, 256, kernel_size=3, padding=1)**: A second convolutional layer with 256 filters to extract higher-level features.
- **ReLU Activation**
- **MaxPool2d(kernel_size=2)**: Downsampling by half
- **ConvTranspose2d(256, 128, kernel_size=2, stride=2)**: Transposed convolution for upsampling, restoring the spatial dimensions.
- **ReLU Activation**
- **Conv2d(128, 3, kernel_size=1)**: Final convolutional layer to produce an output with 3 channels for RGB images.

The model was trained using Adam Optimizer with a learning rate of 1e-4. The loss function is a custom loss function with MSE loss and mask loss. The complete equation has been described under Section III B.

III. RESIDUAL CHANNEL ATTENTION NETWORK

A. Architecture

Our motivation is to explore a ResNet-like architecture that aims to learn the residual noise correction term needed to convert a noisy image with anomalies into a clean image while preserving these anomalies. A notable framework in the field of image restoration is the Residual Channel Attention Network model, which focuses on enhancing image resolution to transform low-resolution images into high-resolution counterparts [1].

We implement modifications to the above architecture to experiment its performance on denoising faulty images. The model is composed of multiple Residual Channel Attention Blocks that selectively emphasize channels critical for noise correction. Through training with clean images and a anomaly mask, the network learns to identify and remove noise from non-defective areas.

The overall RCAN architecture is as follows:

- **Input (Noisy Image)** - A noisy image with anomalies is input to the network.
- **Initial Conv (64 7x7 filters, padding=3)** - A convolutional layer with a 7x7 kernel to extract initial features with 64 feature maps from the input image.
- **RCAN Block(s)** - 10 Residual Channel Attention Blocks (RCAB) that process the features applying residual learning and attention mechanisms
- **Final Conv (7x7 filters, padding=3)** - A final convolutional layer to reduce the output channels to 3 matching the input, producing the clean image with retained anomalies
- **Output (Clean Image)** - The final output, a denoised image that preserves the desired anomaly information.

The internal architecture of the RCAN block is as follows:

- **Conv1 (3x3, padding=2, dilation=2)** - Dilated convolution is used to expand the receptive field without increasing the number of parameters
- **ReLU Activation** - Applies a ReLU activation function for non-linearity.
- **Conv2 (3x3, padding=2, dilation=2)**
- **Channel Attention:**
  - **Adaptive Average Pooling** - Computes average pooling across all entries per channel
  - **Conv3 (1x1, reduction = 16)** - Reduces the channels to $\frac{C}{16}$. This is analogus to dimensionality reduction and helps to retain only the critical channels for noise correction.
  - **ReLU Activation** - ReLU activation for non-linearity within the attention mechanism.
  - **Conv4 (1x1, expansion to original C)** - Expands channels back to the original number, providing per-channel scaling factors.
  - **Sigmoid Activation** - Applies sigmoid activation to obtain attention weights between 0 and 1.
- **Channel Attention Application** - Element-wise multiplication of the attention weights with the output of Conv2.
- **Skip Connection** - Add the residual connection by summing the input and the output of the attention module.

The detailed architecture can be visualized using Fig. 1.

B. Training Details

The output obtained from the network is trained using a custom loss function against the clean image and the anomaly mask. The custom loss function is defined as

$$\text{Loss} = \alpha \cdot \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 + (1-\alpha)\cdot\frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2 \cdot \text{def\_mask}_i \tag{1}$$

The first term, weighted by $\alpha = 0.3$, is a standard MSE loss that encourages overall denoising across the image. The second term, weighted by $1 - \alpha = 0.7$, applies MSE only to defective regions using the anomaly mask, reinforcing preservation of anomaly details. This balance ensures that the network focuses more on retaining anomalies while still reducing noise in non-defective areas. We train the model for 350 epochs using Adam optimizer and a learning rate of $1e^{-4}$

C. Our modifications to the architecture

We have made slight modifications to the original RCAN architecture to optimize its performance. Our primary contribution is the formulation of our loss function, which integrates both Mean Squared Error (MSE) and masked anomaly loss. Additionally, we have selected custom filter sizes and implemented dilation techniques to enhance the receptive field.

IV. Training and Validation Results

The important metrics used to compare various models that were explored are the PSNR and the SSIM. 29dB was set as a benchmark for the average validation PSNR after going through PSNR statistics in [3]. While the UNet model failed to cross this threshold, the RCAN model performed really well. An **average PSNR of 29.94dB** was achieved on the validation set, with a PSNR of 30.33dB on the training set. The SSIM achieved on the validation set was 0.805. Testing the model on images of some objects like the ”metal nut” and ”pill” gave significantly high PSNR values of 34.68dB and 33.47dB respectively. Shown below in Fig. 4 and Fig. 5 are the object-wise statistics of the two trained models. All PSNR values are in dB. These numbers above are supported by some clean de-noised images with the anomalies still intact shown below in Fig. 2.

Another observation that can be made is of the anomaly-preserving nature of our model. In case of models which are trained to regenerate images from noise, we may lose the features of anomalies. In order to observe this, we have plotted the Mean Square Error of a generated image with respected to its original image (Fig. 3). As we can see, the MSE is uniform throughout and there is not much deviation at the points where anomaly is present in the original image. Moreover, we can also see these anomalies clearly in the generated images in Fig. 2. This shows that the model did not accidentally patch up the anomaly and thus, anomalies can be detected in the reconstructed images.

V. Conclusion

RCAN and U-Net architectures were trained and tested on the validation set. The two models have been compared and a significant advantage of using a Residual model for de-noising has been shown. We achieved a PSNR of 29.94dB and an SSIM of 0.805 on the validation set. We were also able to preserve anomalies and verified this by plotting various MSE plots of different images generated with respect to their ground truths.
Attached is our GitHub project: Project Link

References

[1] J Chen et al., ”Three-dimensional residual channel attention networks denoise and sharpen fluorescence microscopy image volumes” https://github.com/AiviaCommunity/3D-RCAN

[2] J Gurrola-Ramos et al., ”A Residual Dense U-Net Neural Network for Image Denoising” https://github.com/JavierGurrola/RDUNet

[3] J Liang et al., ”SwinIR: Image Restoration Using Swin Transformer” https://github.com/JingyunLiang/SwinIR/tree/main

Images

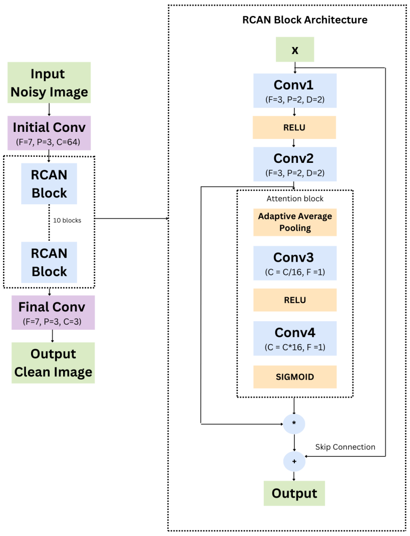

Fig. 1. The architecture used for the RCAN model

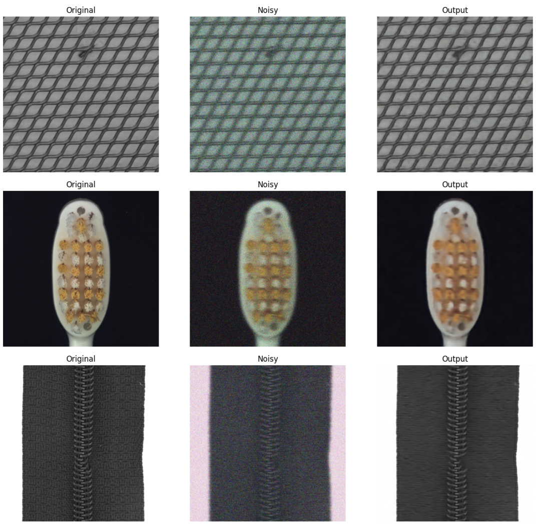

Fig. 2. A few clean, noisy and generated image triplets

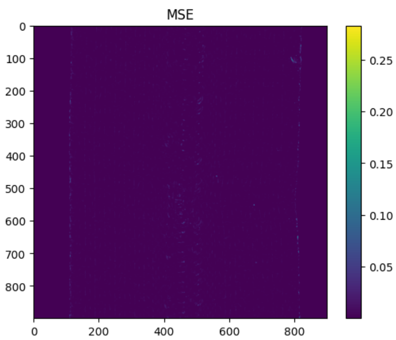

Fig. 3. The Mean Squared Loss of the generated image with respect to the original image of the last example in Fig. 2

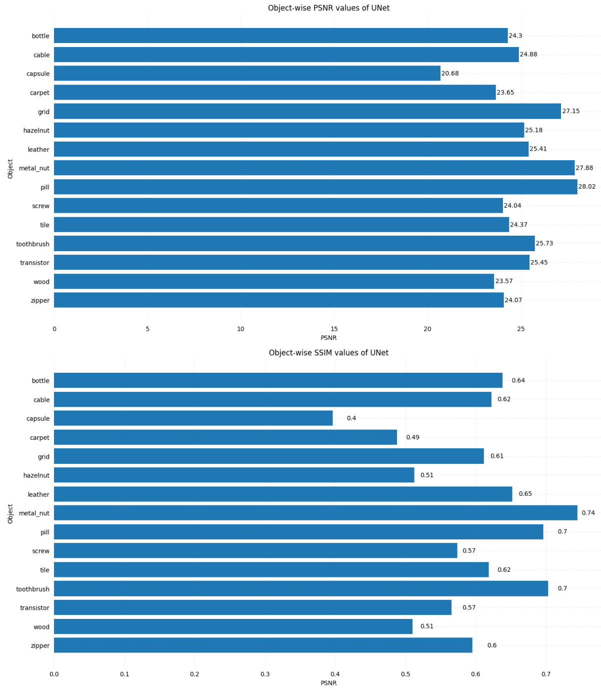

Fig. 4. The object-wise PSNR and SSIM of UNet

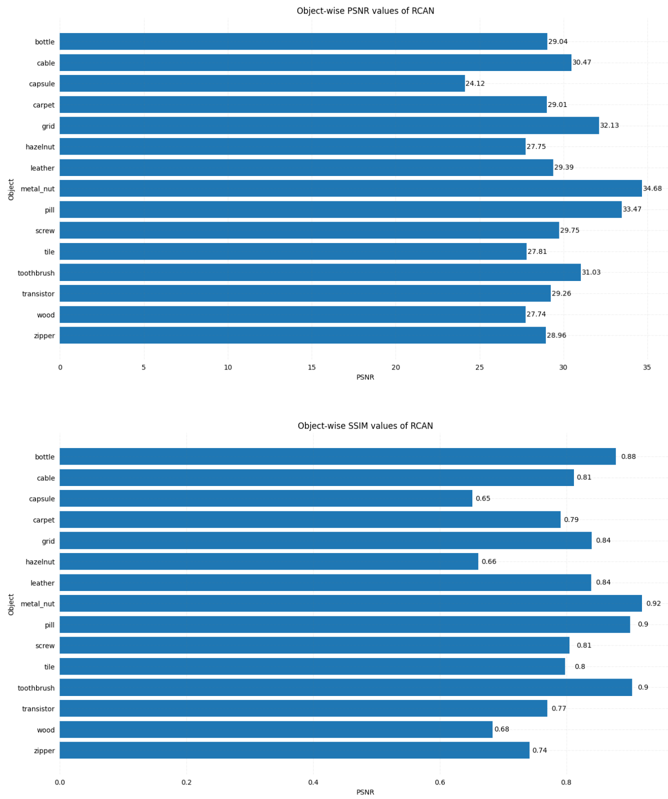

Fig. 5. The object-wise PSNR and SSIM of RCAN