

Restoring An Image Taken Through a Window Covered with Dirt or Rain

Supplementary Material

David Eigen Dilip Krishnan Rob Fergus

Dept. of Computer Science, Courant Institute, New York University

{deigen,dilip,fergus}@cs.nyu.edu

1. Dataset Colleciton

1.1. Synthetic Dirt Extraction

To find α and αD , we took pictures of several backgrounds displayed on a projector screen, both with and without a dirt-on-glass pane placed in front of the camera (see Fig. 1(top)). Because we used a projector to switch backgrounds and did not move the camera, the resulting images were pixel-aligned, and thus yielded multiple examples of each pixel under dirt and non-dirt conditions. We then solved a least-squares system to find the values for α and αD at each pixel. Given captured image pairs $\{(I_k, I'_k)\}_{k=1}^K$, $K \geq 4$, we solve the system of K equations implied by our generation model at each pixel location (i, j) :

$$I'_k(i, j) = \alpha(i, j)D(i, j) + (1 - \alpha(i, j))I_k(i, j), \quad k = 1, \dots, K$$

In our setup, we projected backgrounds of solid white, red, green and blue. We also illuminated the dirt directly using a spotlight, to reduce effects of backlighting from the projector and to help shorten exposure time.

1.2. Rain Dataset Collection

For corrupt images, we simulated the effect of rain on a window by spraying water on a pane of anti-reflective MgF_2 -coated glass placed between the camera and the scene, taking care to produce drops that closely resemble real rain. Using the tripod setup shown in Fig. 1(bottom), we first took one picture with a clean piece of glass in place (A), then swapped the glass for one with water (B). We reduced single-pixel-scale differences by downsampling the resulting images by a factor of 2. This setup captured pixel-aligned image pairs that we used for training.

Although the time between captures for each pair was fairly short (only several seconds), there were global illumination changes that caused an approximate mean shift between corresponding clean and rain images. We corrected for these by scaling the clean image in each pair by a single

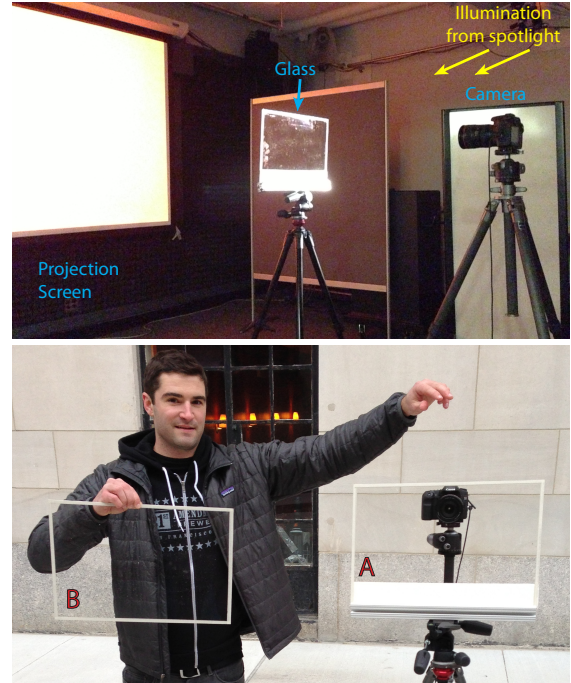


Figure 1. Training data capture setups for dirt (top) and water drops (bottom).

constant value, chosen to minimize average error between it and the corresponding noisy image. In addition, it was essential to minimize object motion between corresponding images, in order for their difference to be limited to the difference in corruption. We addressed this by using pictures of mostly-static scenes for the training set. Although not perfect, differences due to motion were limited and non-systemic enough to work for training.

Despite the fact that our network was trained on mostly-static scenes, it still preserves the structure of animate parts of the test images fairly well: The face and body of the subject are reproduced with few visible artifacts, as are the tree branches and plants (which move from wind).

2. Rain Sequence

Below are four frames from a time series of rain falling on an initially clean pane of glass. We also include a video of this sequence, `rain-sequence.mov`.

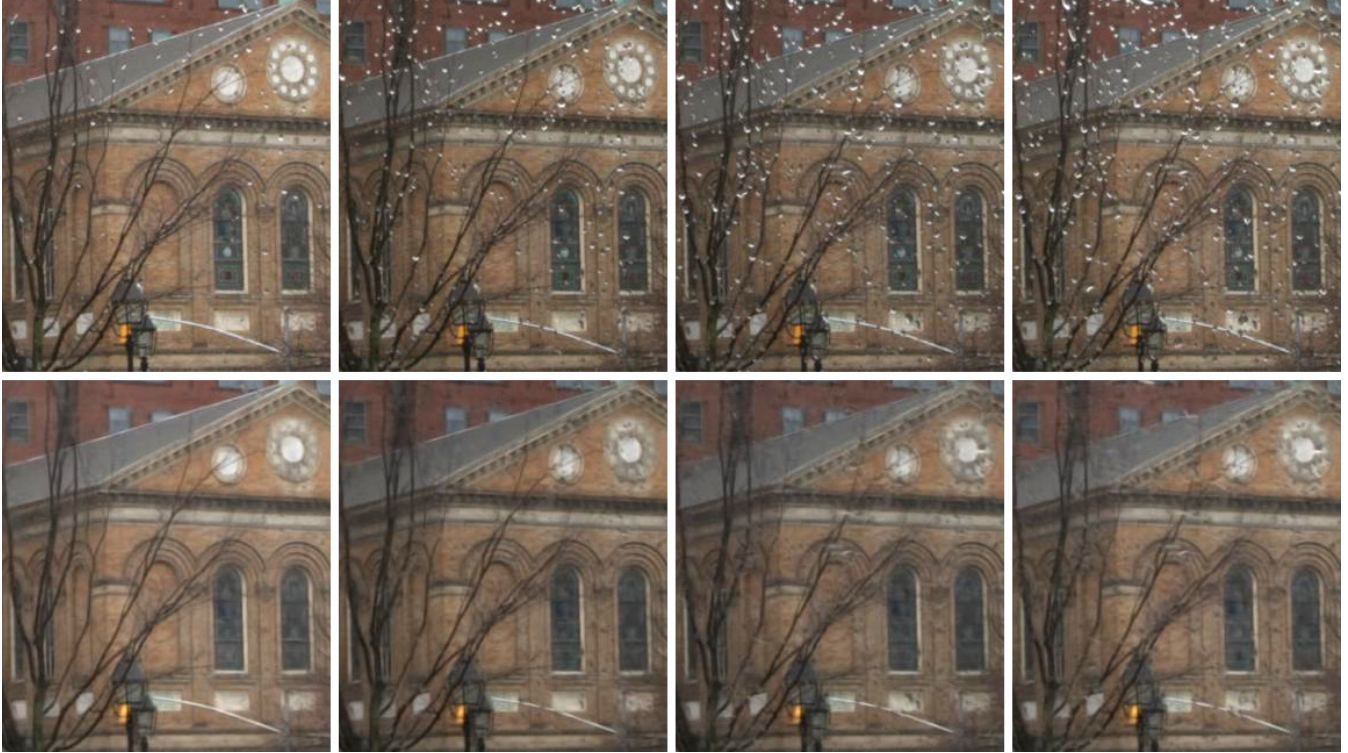


Figure 2. Four shots from the rain video sequence (see the supplementary video), along with the output of our network. Note that each frame is processed independently, with no temporal information or background subtraction of any kind used.

3. Additional Examples

We have included full JPEG images of several results in the `images` directory accompanying this PDF, including all examples in the main paper. Selected image regions from each example are on the following pages.

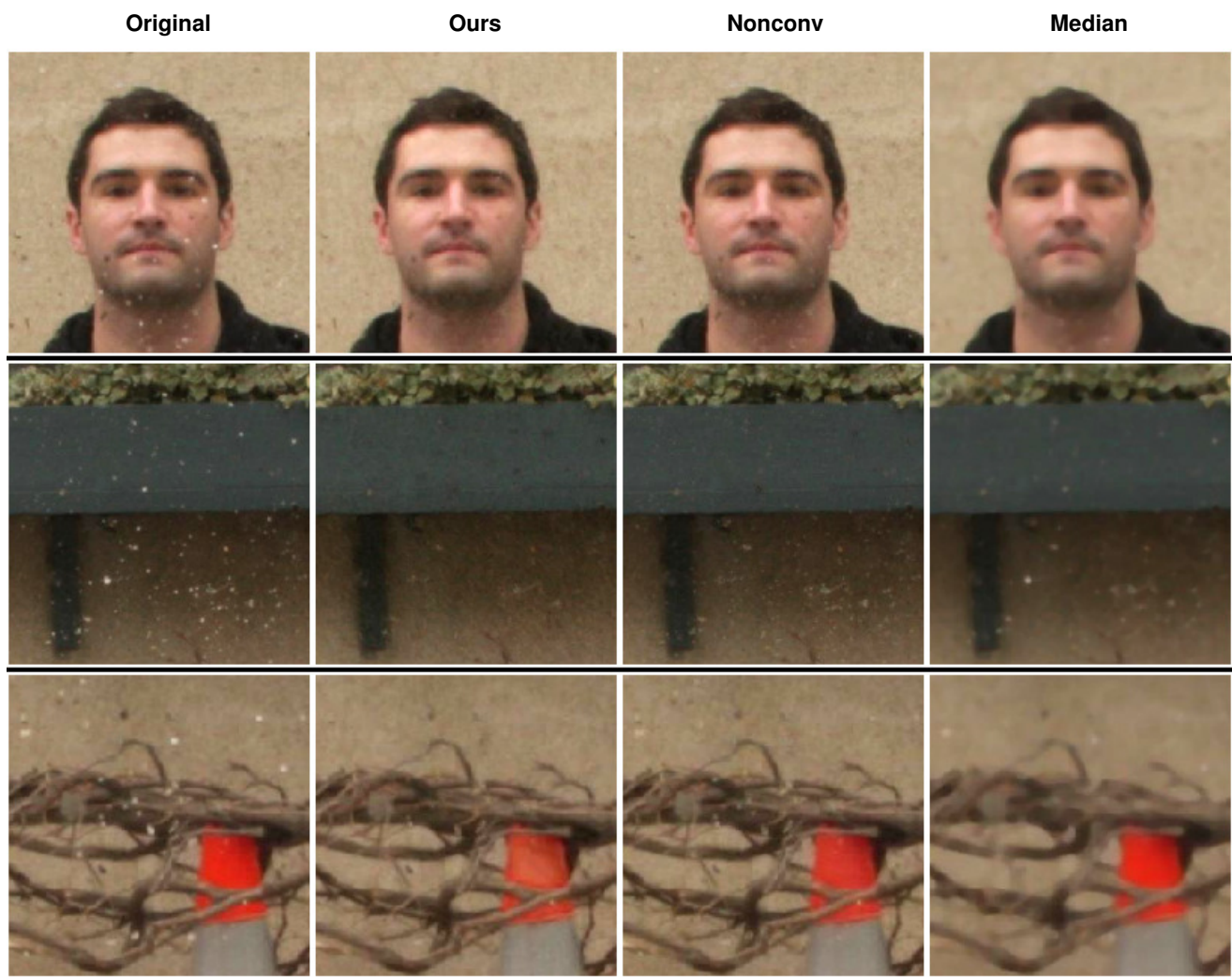


Figure 3. dirt-1 example.

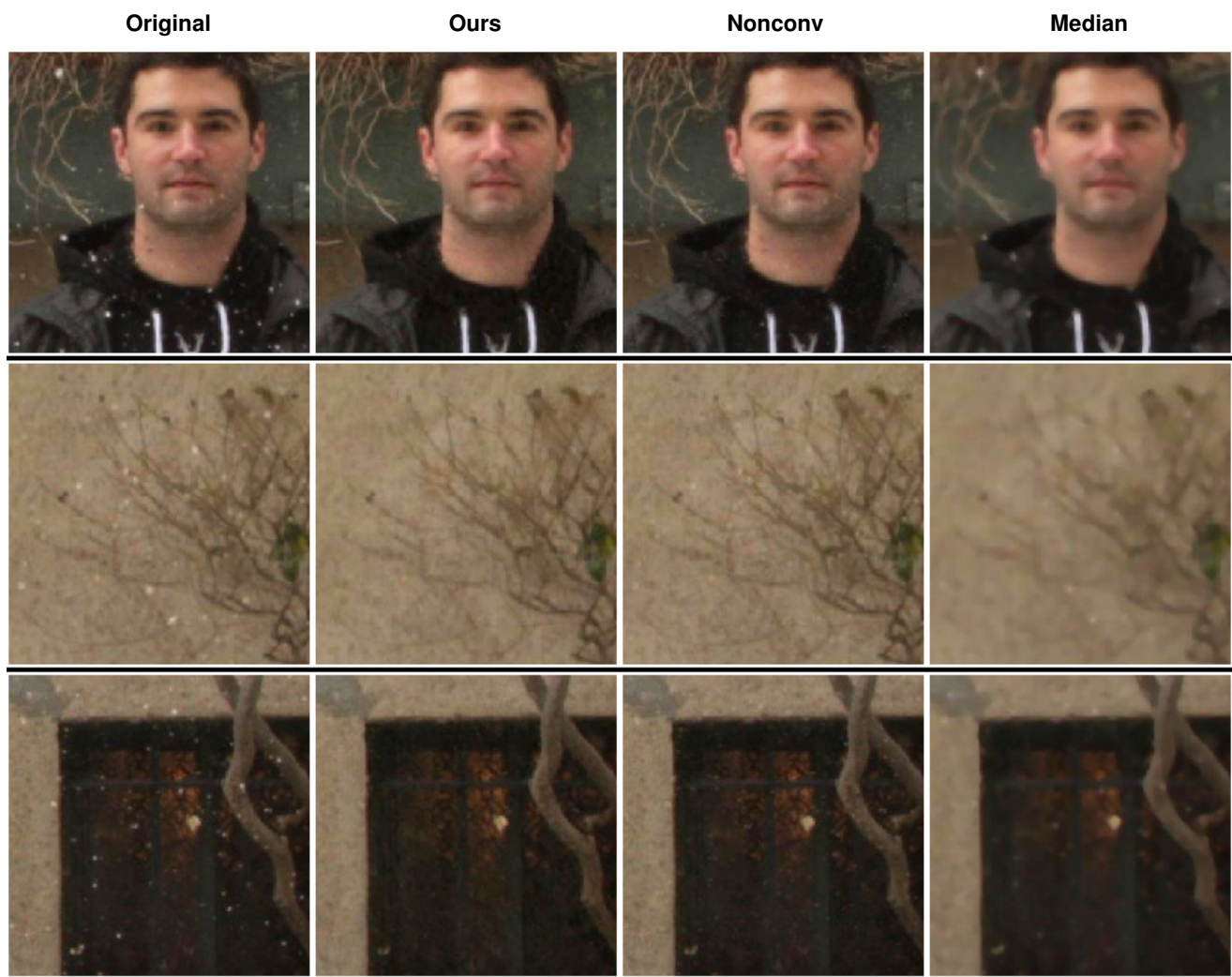


Figure 4. dirt-2 example.



Figure 5. dirt-3 example.

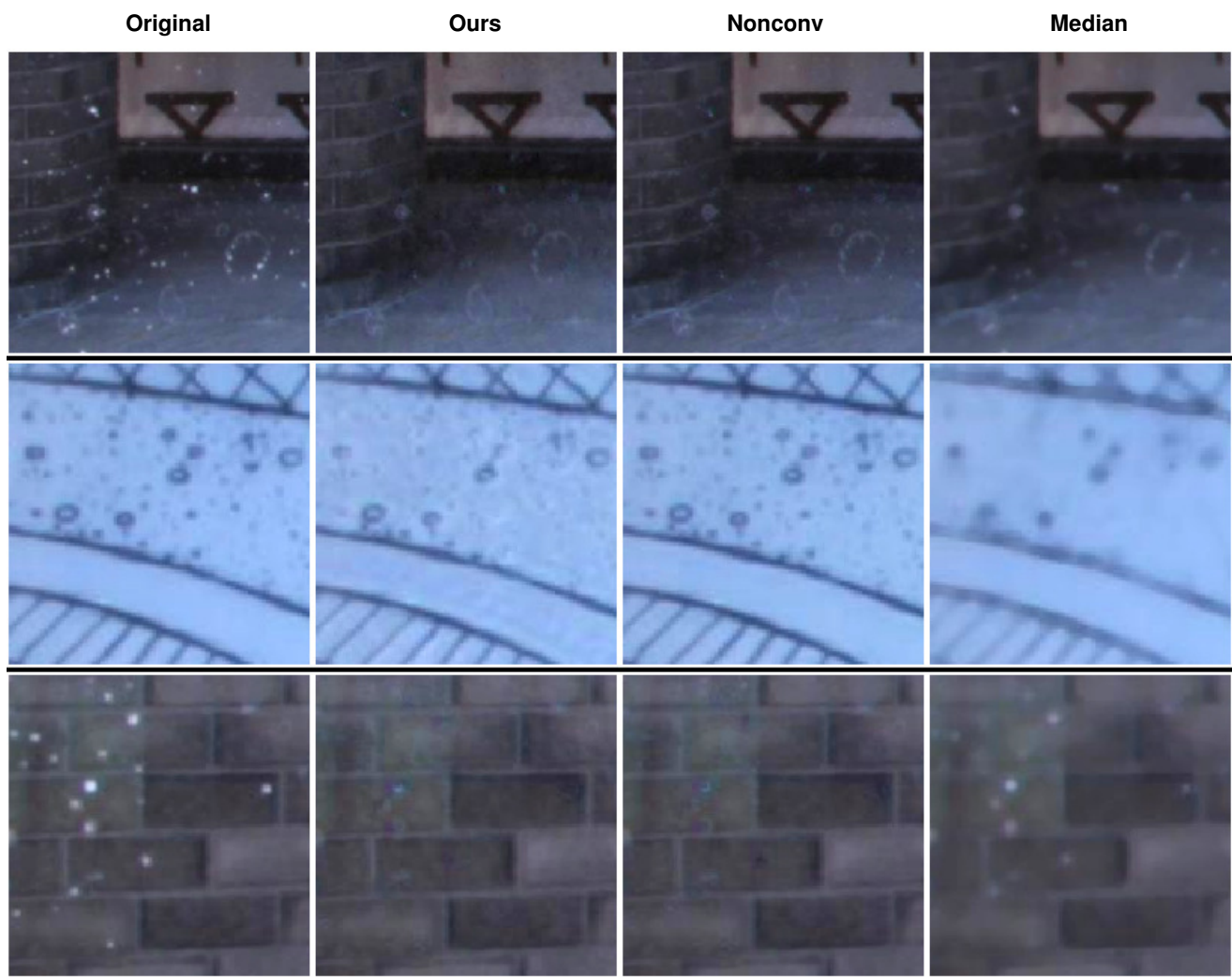


Figure 6. dirt-4 example.

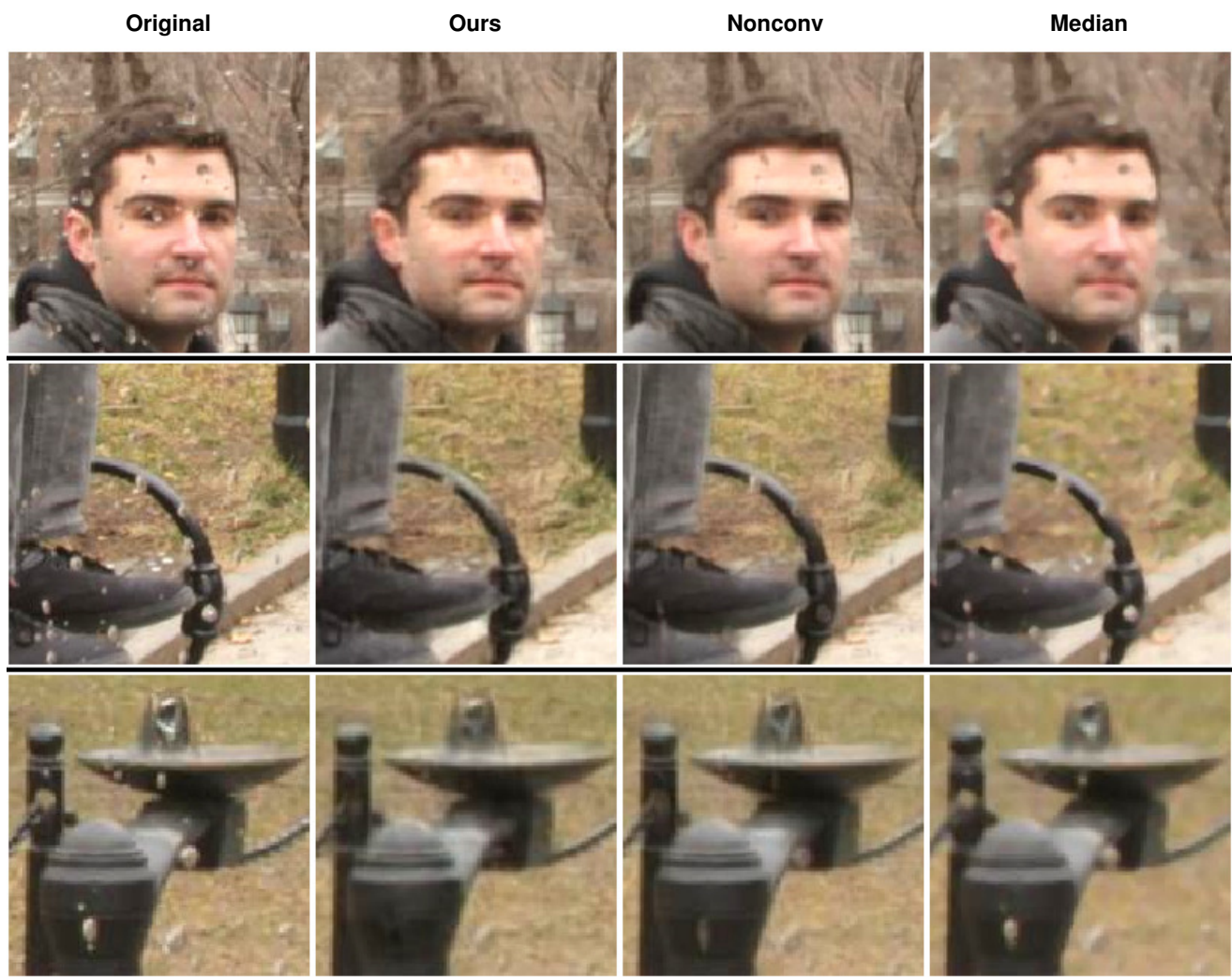


Figure 7. rain-1 example.



Figure 8. rain-2 example.

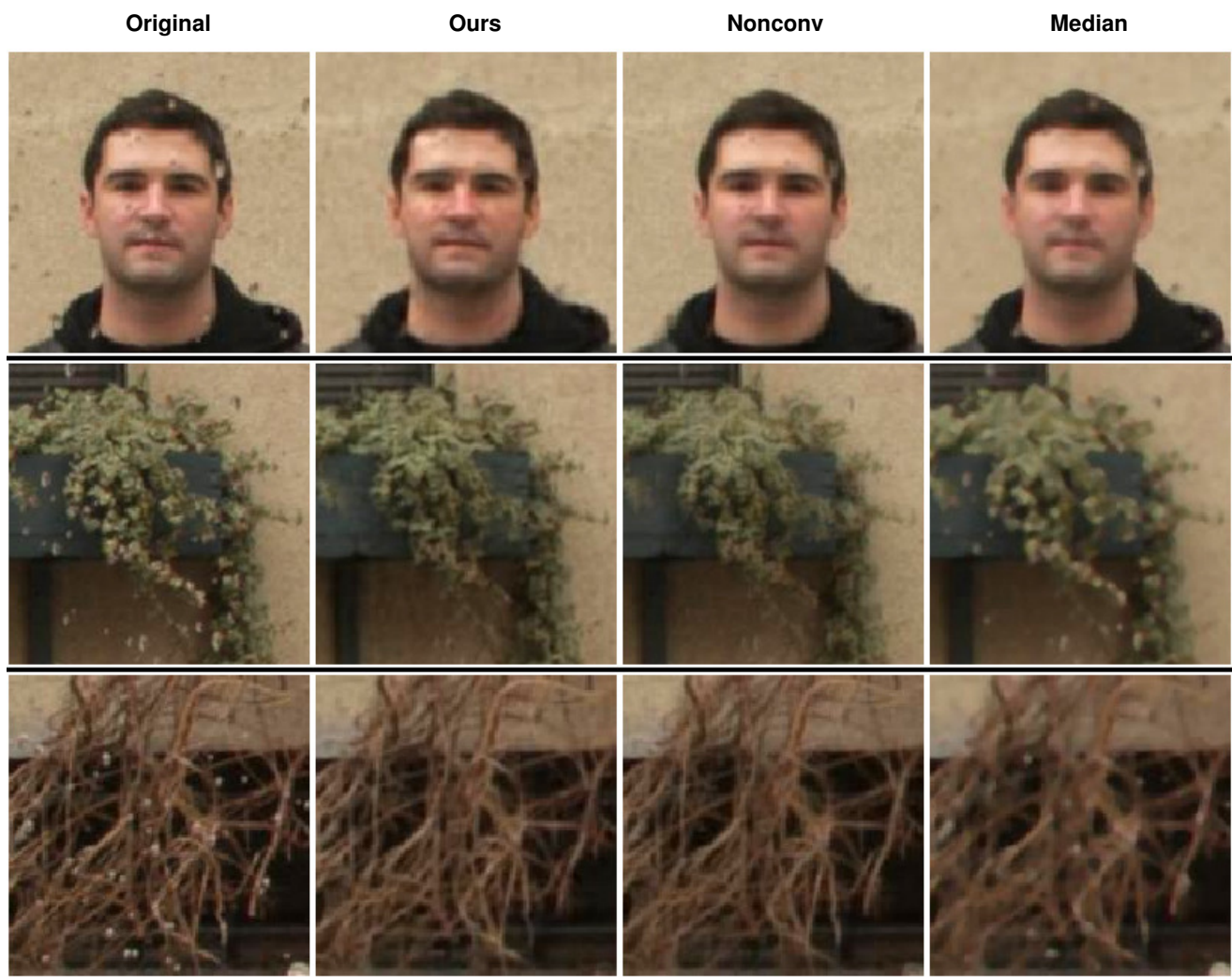


Figure 9. rain-3 example.

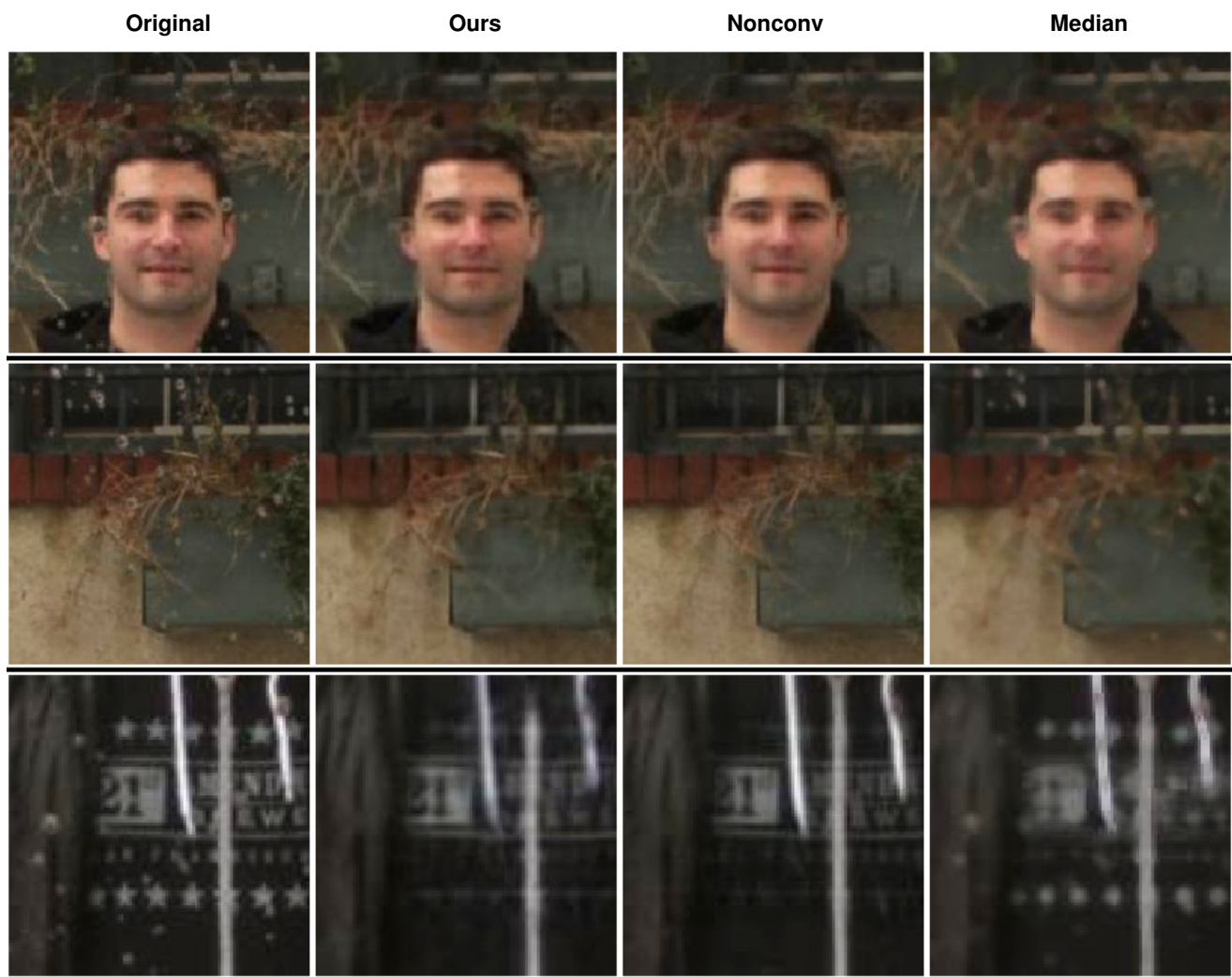


Figure 10. rain-4 example.



Figure 11. rain-5 example.