From Noise to Readable Images: Eavesdropping on Computer Screens via Custom Hardware and Deep Learning Image Reconstruction

Filip Tuch Richard Ostertág

Department of Computer Science, Faculty of Mathematics, Physics and Informatics, Comenius University, Bratislava, Slovakia

26.11.2025

Image Transmission

A computer's image is sent to the monitor as a stream of RGB pixels.

- pixel's R, G and B values are mapped to voltages,
- these voltages are transmitted through a cable,
- in total, each second $x_t \cdot y_t \cdot f_v$ pixels are transmitted,
    - x_t – display width, y_t – display height, f_v – display frame rate.

Van Eck's Radiations

- rapid changes in voltages → electromagnetic emissions,
  - can be captured from distance with a software defined radio (SDR),
- pixel transmission period $t_p = 1/(x_t \cdot y_t \cdot f_v)$,
  - impulses in the captured signal at multiples of t_p,
  - impulse amplitude → grayscale value of the pixel,
- automatic reconstruction with TempestSDR,
  - open-source software, developed by Martin Marinov,
  - offers real-time image reconstruction.

TempestSDR Example Attack

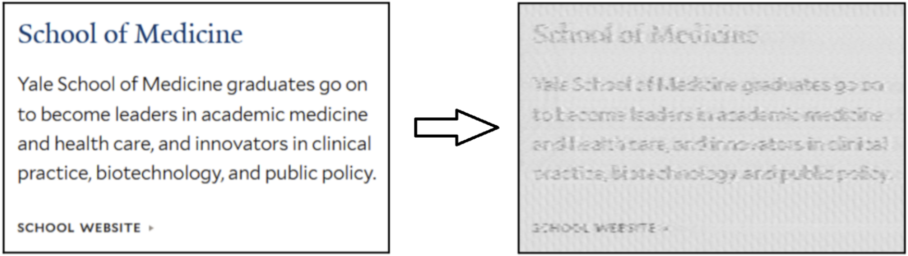

Figure 1: Left to right: original image, reconstructed image in TempestSDR.

Custom Hardware

With standard hardware, the attack range is limited to 1 meter.

1. 5-element Yagi-Uda antenna tuned to 130 MHz,
2. low-noise amplifier for signal gain,
3. band-pass filter (125-135 MHz) for signal isolation.

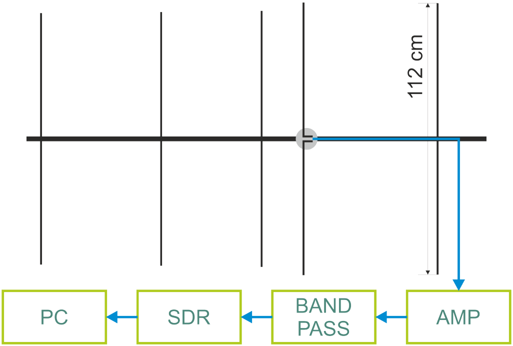

Figure 2: Our Yagi-Uda system.

Custom Hardware

Figure 3: Reference image.

Custom Hardware

Tested in a hallway with a metal structure.

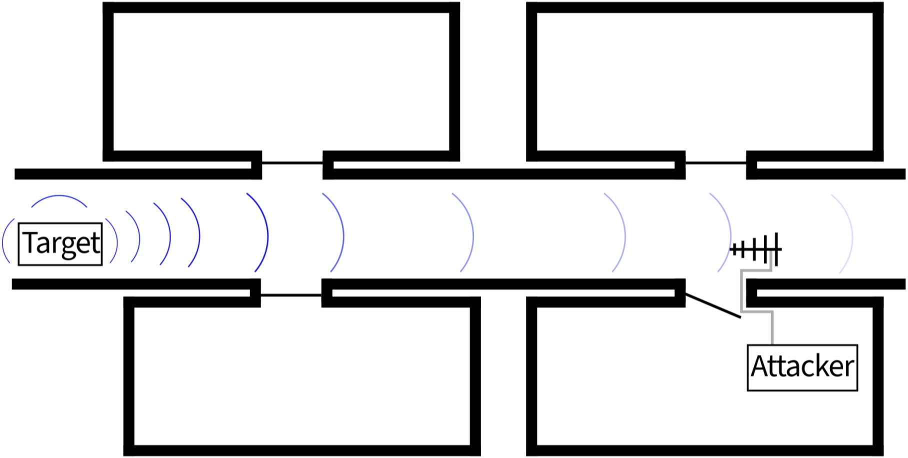

Figure 4: Attack scenario.

Custom Hardware

Figure 5: Captured image from various distances.

Deep Learning Image Reconstruction

Captured images still remain noisy.

- we created a dataset of common computer usages,
  - web pages, text documents, OS interfaces, ...
  - 600 real images + 250 generated images,
  - each image captured ~ 6 times with TempestSDR,
  - automated with PyAutoGUI, Selenium, OpenCV,
- trained two deep learning models for image reconstruction,
  - auto-encoder (Deep Residual U-Net, DRUNet),
  - noise-predictor (Denoising CNN, DnCNN).

Deep Learning Image Reconstruction

Figure 6: An illustrative sample from our dataset.

Deep Learning Image Reconstruction

- PSNR (Peak Signal-to-Noise Ratio),
- SSIM (Structural Similarity Index Measure),
- CER (Character Error Rate).

Method	**PSNR**	**SSIM**	**CER**
TempestSDR	8.65 dB	0.41	86%
DnCNN	17.63 dB	0.75	54%
DRUNet	**19.08 dB**	**0.87**	**46%**

Table 1: Image quality comparison of captured (TempestSDR) and reconstructed (DRUNet and DnCNN) images.

Image Reconstruction Example

Figure 7: Example of the attack.

Image Reconstruction Example

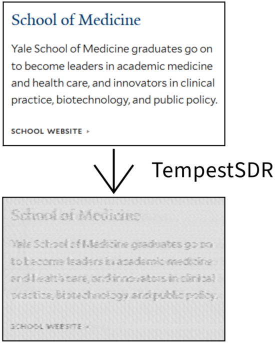

Figure 8: Example of the attack.

Image Reconstruction Example

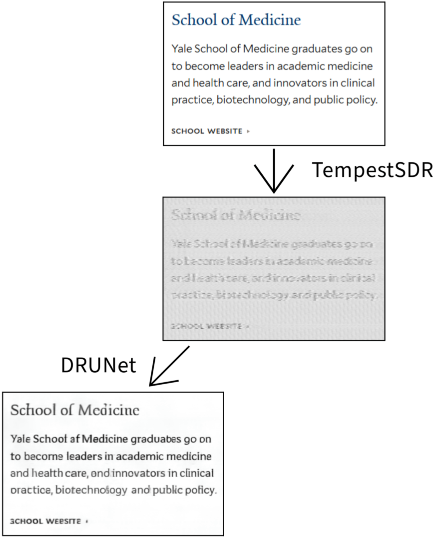

Figure 9: Example of the attack.

Image Reconstruction Example

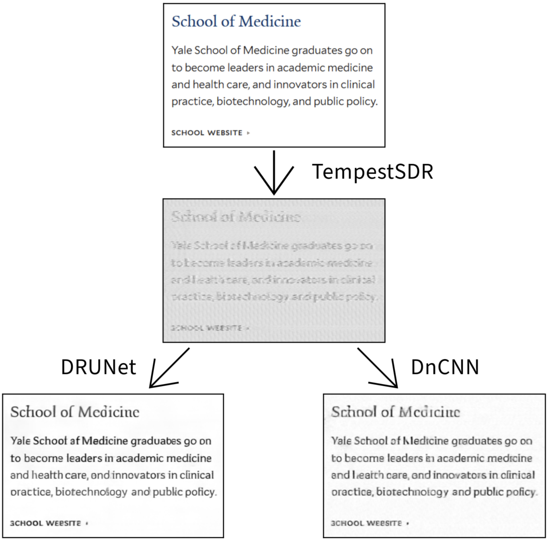

Figure 10: Example of the attack.

Thanks for your attention!