Decoding HDMI Emissions with Deep Learning: Enhancing TEMPEST Eavesdropping on Digital Displays

November 2024

Federico Larroca

FACULTAD DE INGENIERÍA

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

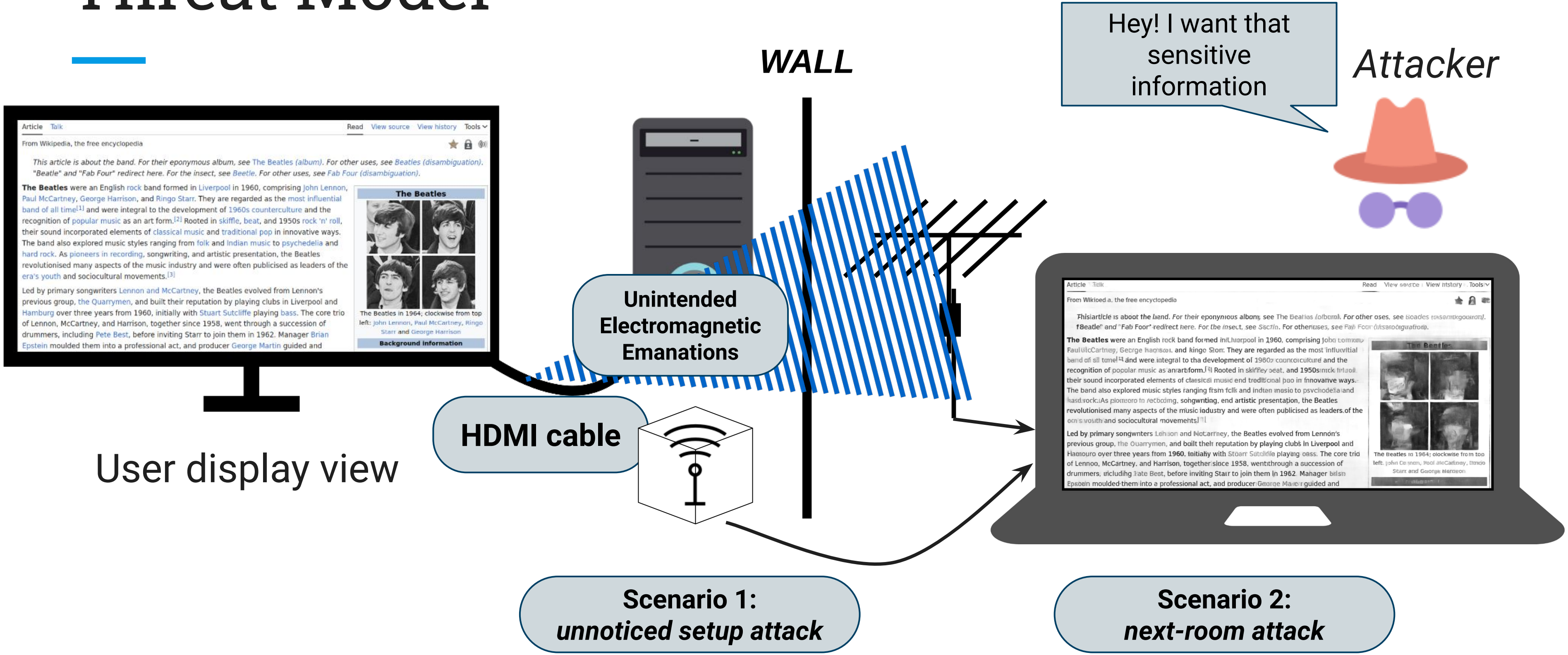
Acknowledgements

This is a long-standing research line, and I’ve had the pleasure of collaborating with several colleagues:

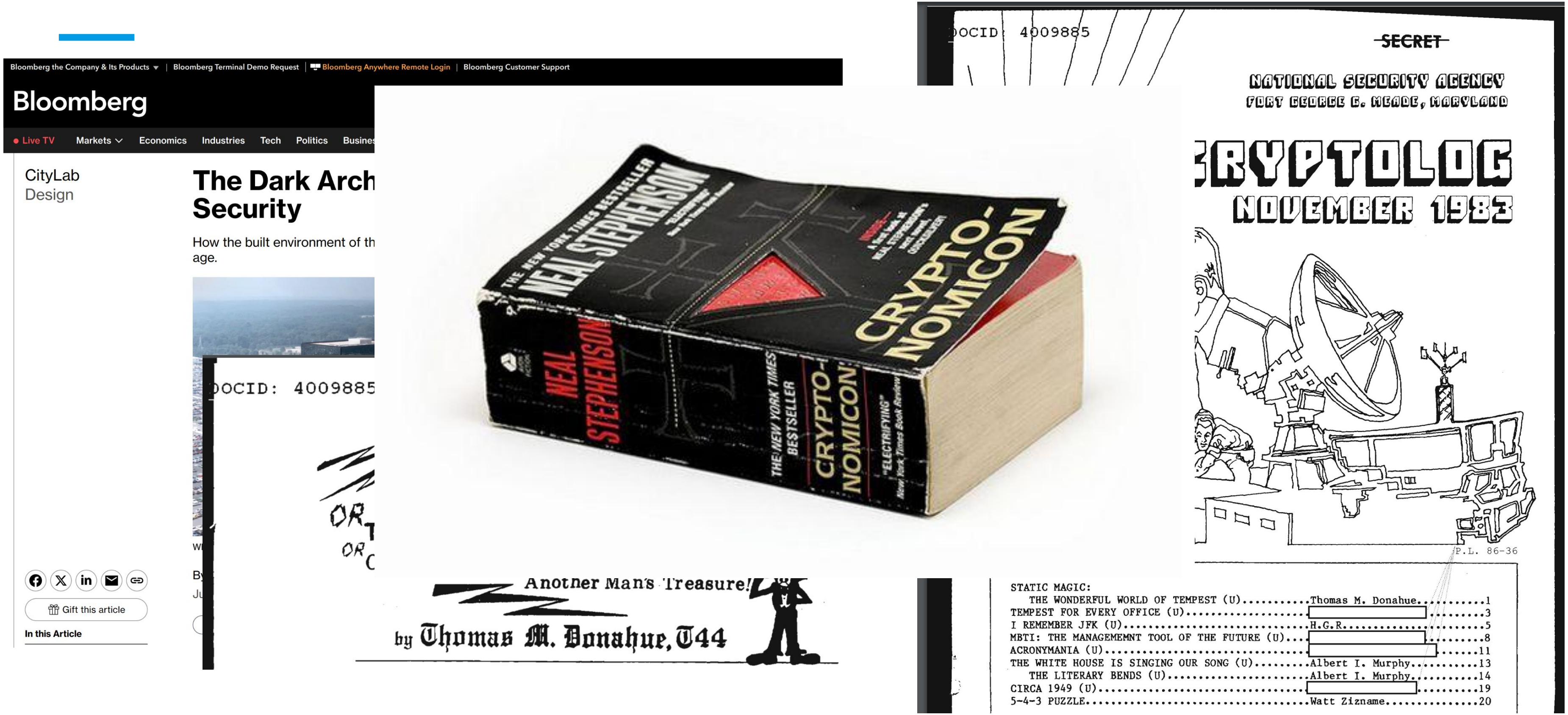
- Pablo Menoni
- Pablo Bertrand, Felipe Carrau and Victoria Severi
- Emilio Martínez, Santiago Fernández and Gabriel Varela
- Pablo Musé

What is ***TEMPEST***? Why ***Deep***?
Demo Video:

Threat Model



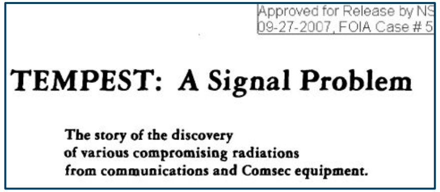
But wait… is this even a thing?



TEMPEST / Van Eck Phreaking

1939-1945

World War 2
Unintended emanations from teleprinter signals


Approved for Release by NS
09-27-2007, FOIA Case # 5

TEMPEST: A Signal Problem

The story of the discovery of various compromising radiations from communications and Comsec equipment.

1985

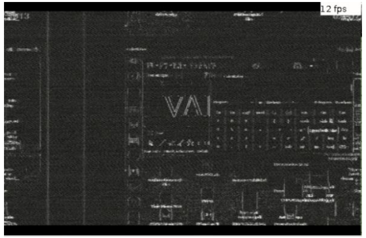
Wim Van Eck
Cathode Ray Tubes attack experiments

2003

Markus Kuhn
VGA, DVI and HDMI wideband receiver

UNIVERSITY OF CAMBRIDGE

2015



Martin Marinov
TempestSDR
Java-based framework

2018

Pablo Menoni
Experiments with *TempestSDR*
Analytical formulations

2020

Federico Larroca
gr-tempest
GNU Radio framework for Tempest

2021

Bertrand P., Carrau F., Severi V.
gr-tempest 2.0
Improving functionalities of *gr-tempest*

UNIVERSIDAD DE LA REPÚBLICA URUGUAY

2024


Source: *gr-tempest 2.0*

deep-tempest

Why are we doing this?

- **Improve results** from **previous works** of colleagues for HDMI cables, using **deep learning**.

- Test **robustness** and find **countermeasures**.

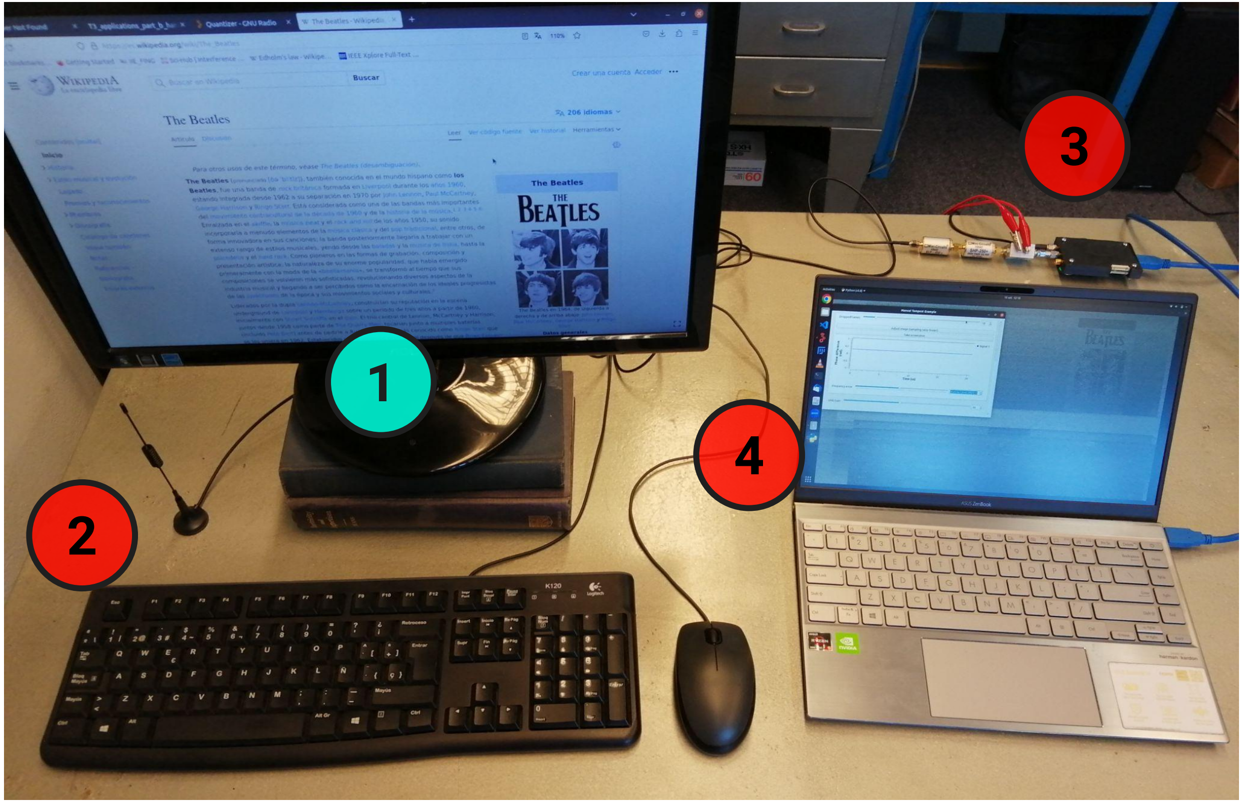
- Because, as we can, ***others could too…***

RAISE AWARENESS

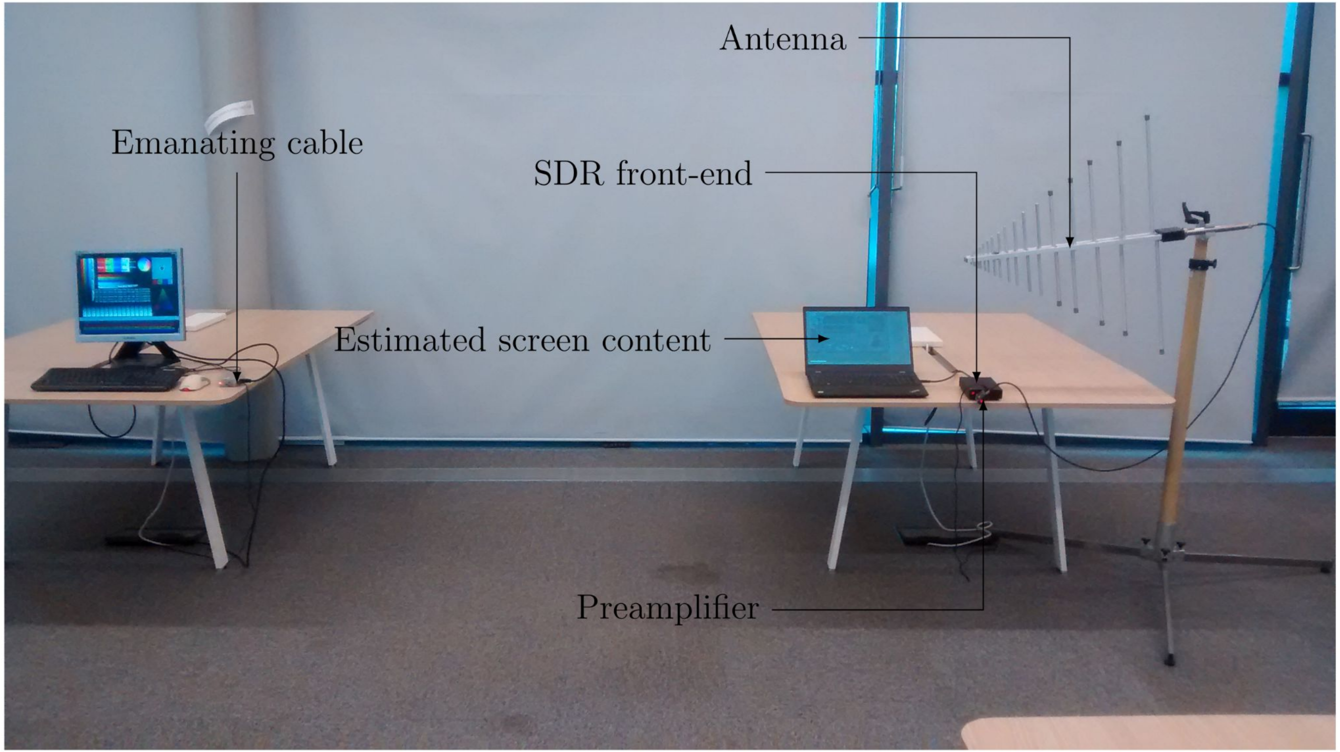
Our setup

1) Display with HDMI to spy on *(1600x900@60 fps)*
2) Antenna
3) SDR and RF filters
4) Spying PC, running ***deep-tempest***

green→users PC ***red → attacker***

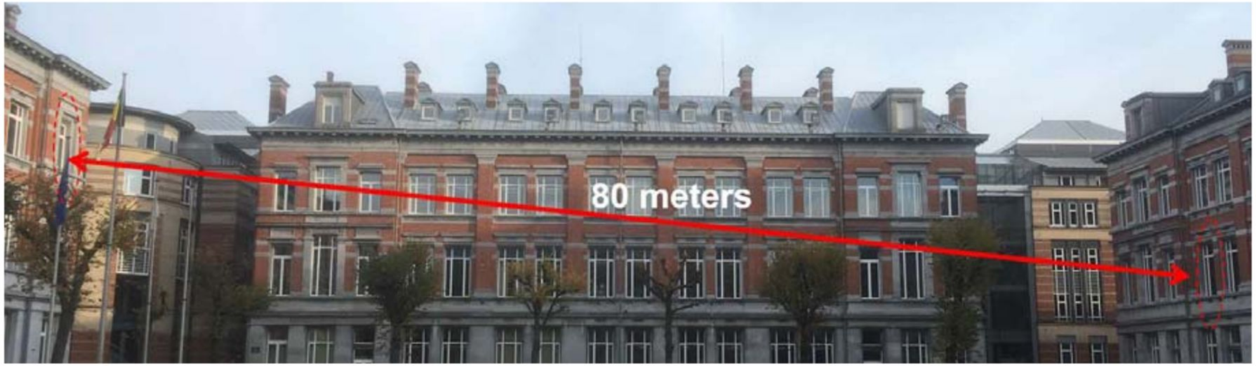


Other setups

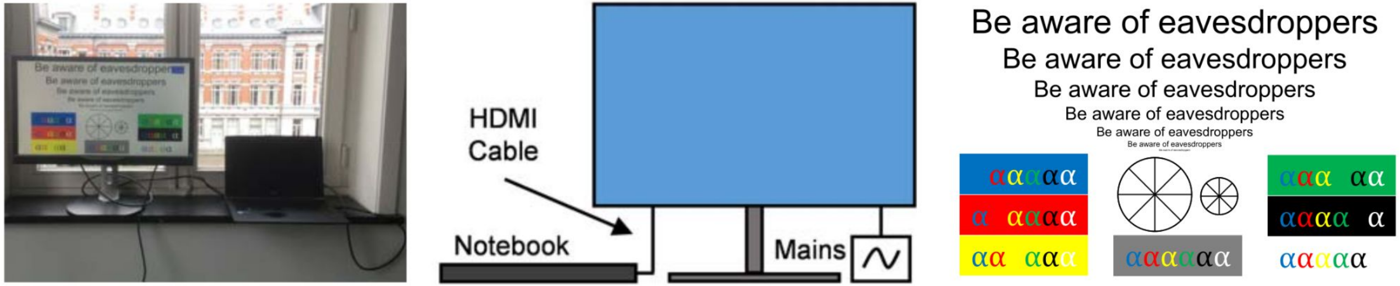


Christian D. O’Connell, “Exploiting quasiperiodic electromagnetic radiation using software-defined radio”, PhD thesis, Cambridge, 2018

Other setups



(e)

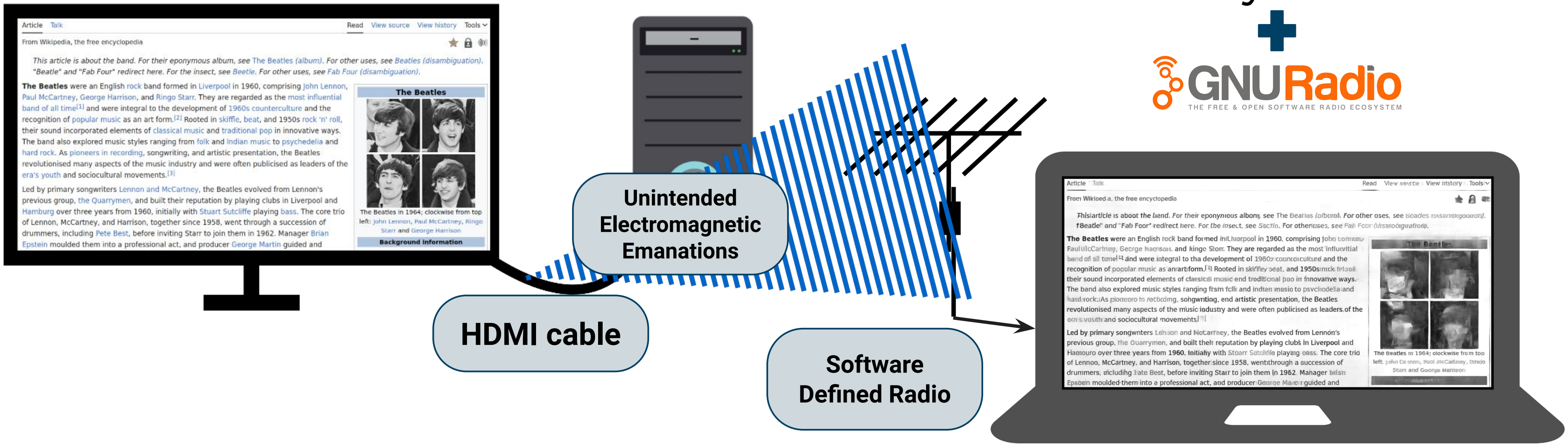


P. de Meulemeester et al., “Eavesdropping a (Ultra-)High-Definition Video Display from an 80 Meter Distance Under Realistic Circumstances”, EMCSI 2020

Image recovery roadmap: electromagnetics

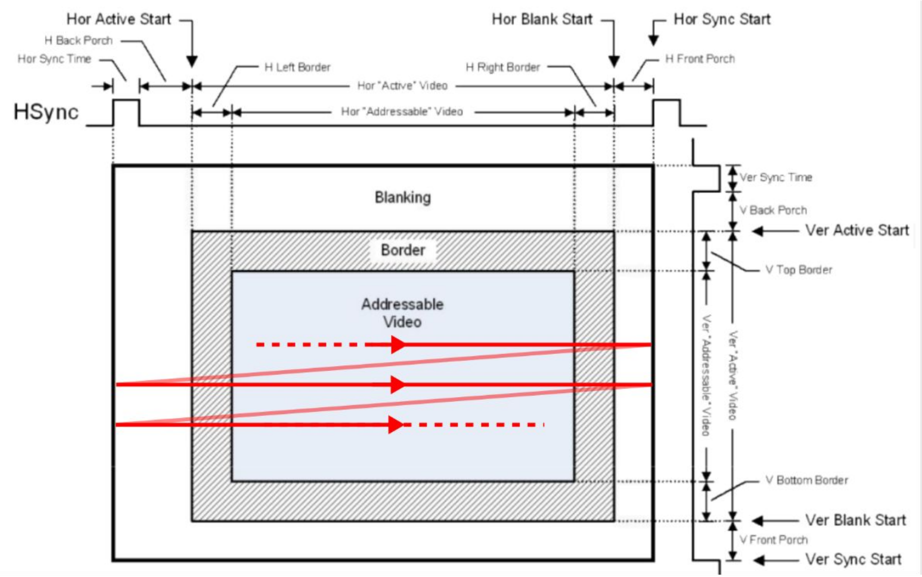
PyTorch

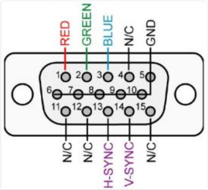
GNU Radio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

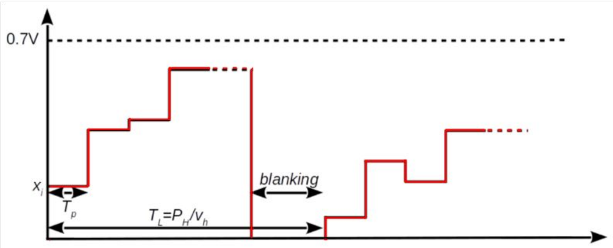


Simpler case: VGA ("analog")

VGA signal: PAM with rectangular pulses (basically a Zero-Order Hold DAC)







Simpler case: VGA ("analog")

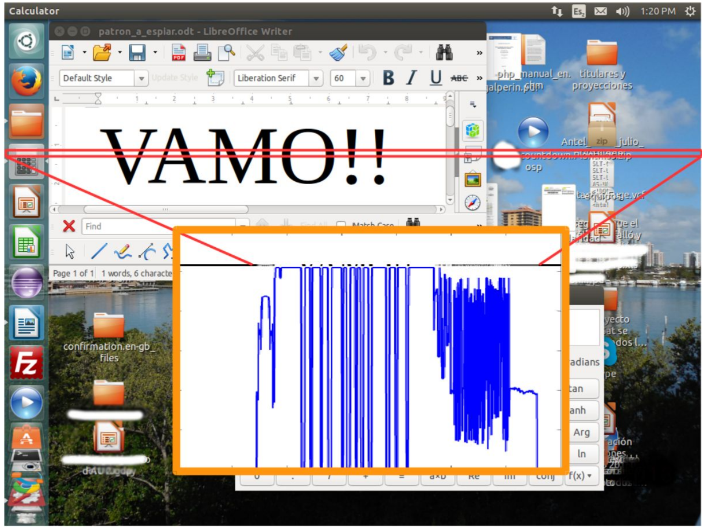
What's the spectrum? A little math:

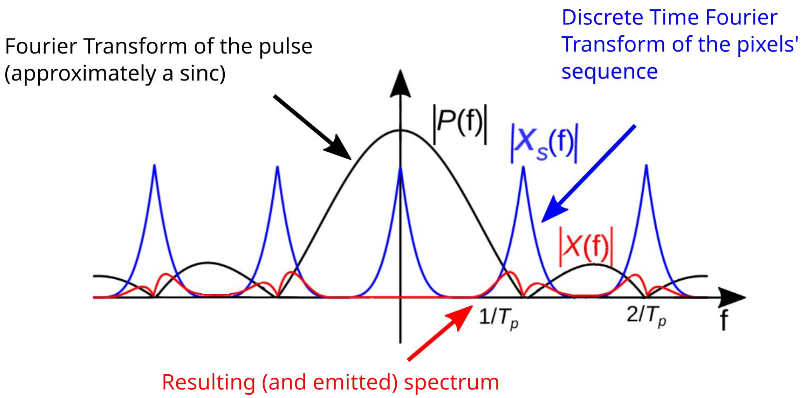
$$x(t) = \sum_i x_i p(t - iT_p)$$

$$\Rightarrow X(f) = \sum_i x_i P(f) e^{-j2\pi f T_p}$$

$$\Rightarrow X(f) = P(f) \sum_i x_i e^{-j2\pi f T_p}$$

$$\Rightarrow \color{red}{X(f)} = P(f) \color{blue}{X_s(f)}$$



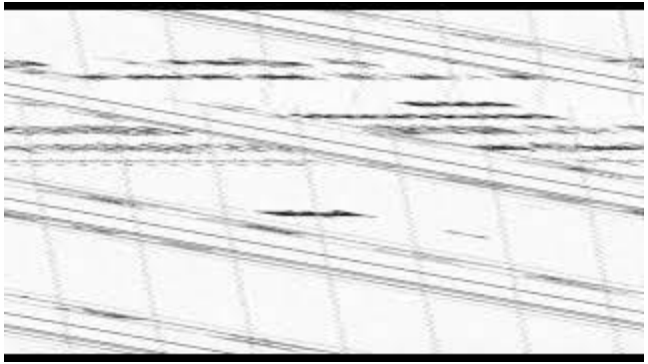


“Demodulating” a VGA signal

Recipe:

1. Point antenna to VGA connectors,
2. Demodulate at a carrier of fc = 1/Tp
   - Ex. 1024×768@60Hz => 1/Tp ≈ 65 MHz
3. Take the samples’ magnitude to avoid frequency synchronization issues

What do I get?

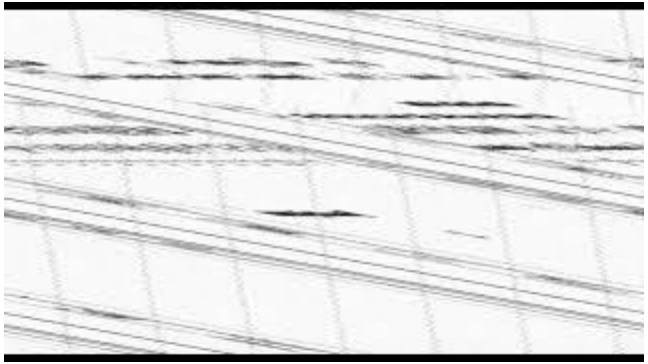


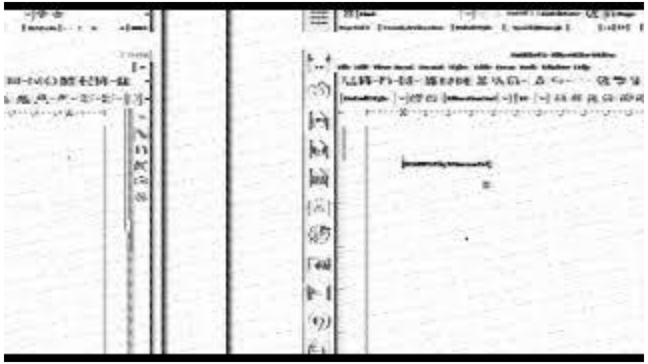
“Demodulating” a VGA signal

Recipe:

1. Point antenna to VGA connectors,
2. Demodulate at a carrier of fc = 1/Tp
   - Ex. 1024×768@60Hz => 1/Tp ≈ 65 MHz
3. Take the samples’ magnitude to avoid frequency synchronization issues

What do I get?



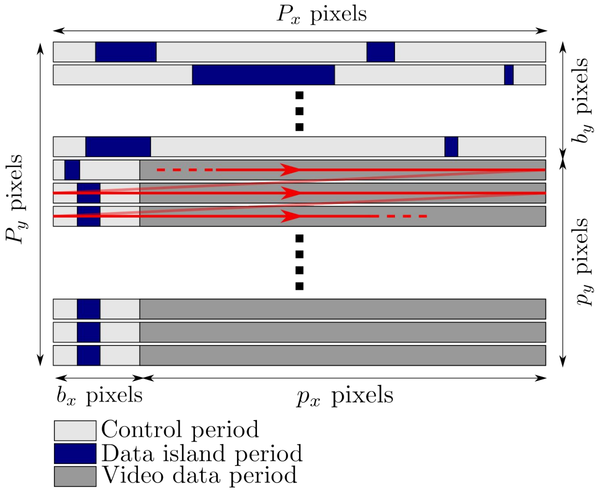


Challenge:

- Time synchronization (all video interfaces have a specific pin for this)
- Repetition is our friend here:
  - One line is similar to the next (coarse)
  - One frame is similar to the next (fine-grained)

Spying on HDMI

Great! What about HDMI? Digital encoding complicates the problem



HDMI Digital signal
Transition Minimized Differential Signaling

Each color intensity (0-255) represented with 8 bits

➔ Mapped to a 10 bit word (non-linear and with memory)

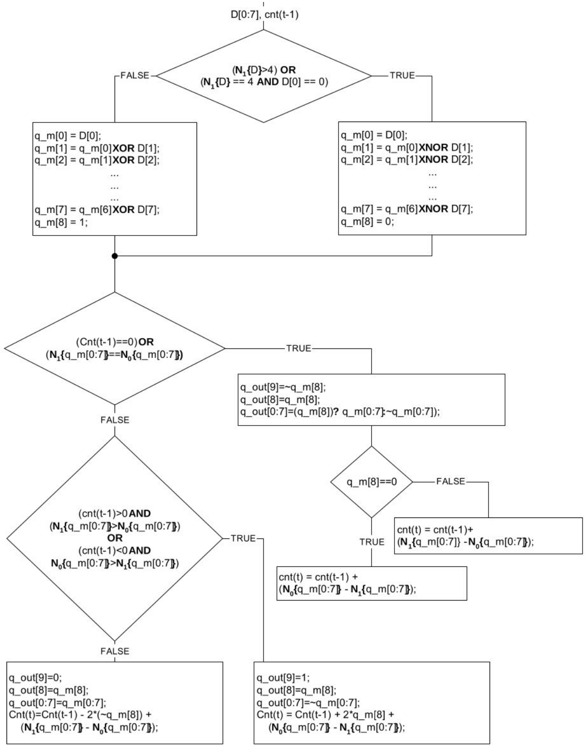
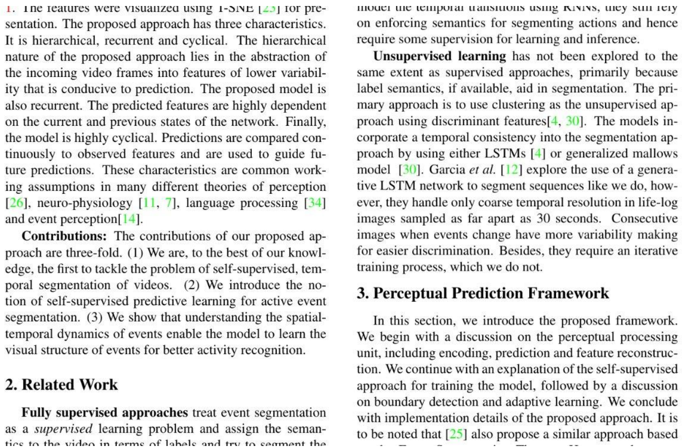


Figure 5-7 TMDS Video Data Encode Algorithm

HDMI Digital signal
Transition Minimized Differential Signaling

Each color intensity (0-255) represented with 8 bits

➔ Mapped to a 10 bit word (non-linear and with memory)



sentation. The proposed approach has three characteristics. It is hierarchical, recurrent and cyclical. The hierarchical nature of the proposed approach lies in the abstraction of the incoming video frames into features of lower variability that is conducive to prediction. The proposed model is also recurrent. The predicted features are highly dependent on the current and previous states of the network. Finally, the model is highly cyclical. Predictions are compared continuously to observed features and are used to guide future predictions. These characteristics are common working assumptions in many different theories of perception [26], neuro-physiology [11, 7], language processing [34] and event perception[14].

Contributions: The contributions of our proposed approach are three-fold. (1) We are, to the best of our knowledge, the first to tackle the problem of self-supervised, temporal segmentation of videos. (2) We introduce the notion of self-supervised predictive learning for active event segmentation. (3) We show that understanding the spatial-temporal dynamics of events enable the model to learn the visual structure of events for better activity recognition.

2. Related Work

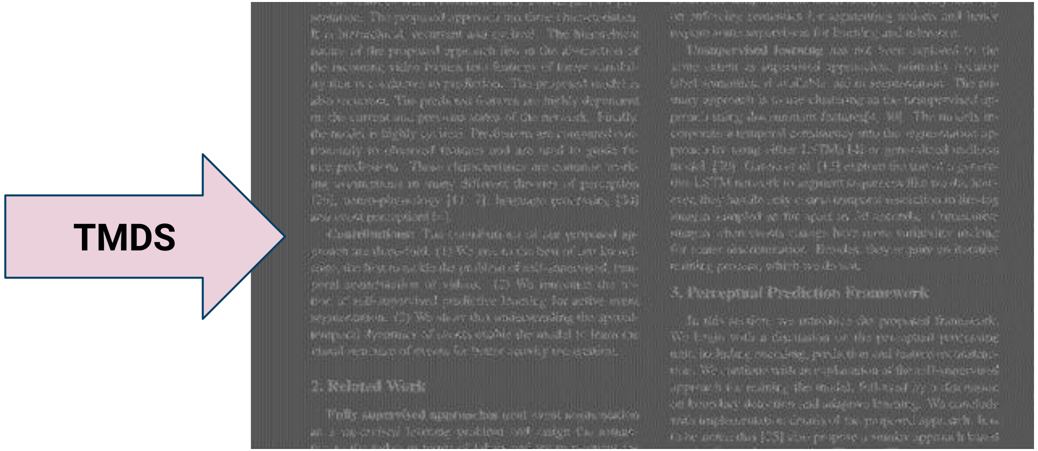
Fully supervised approaches treat event segmentation as a *supervised* learning problem and assign the seman-

on enforcing semantics for segmenting actions and hence require some supervision for learning and inference.

Unsupervised learning has not been explored to the same extent as supervised approaches, primarily because label semantics, if available, aid in segmentation. The primary approach is to use clustering as the unsupervised approach using discriminant features[4, 30]. The models incorporate a temporal consistency into the segmentation approach by using either LSTMs [4] or generalized mallows model [30]. Garcia *et al.* [12] explore the use of a generative LSTM network to segment sequences like we do, however, they handle only coarse temporal resolution in life-log images sampled as far apart as 30 seconds. Consecutive images when events change have more variability making for easier discrimination. Besides, they require an iterative training process, which we do not.

3. Perceptual Prediction Framework

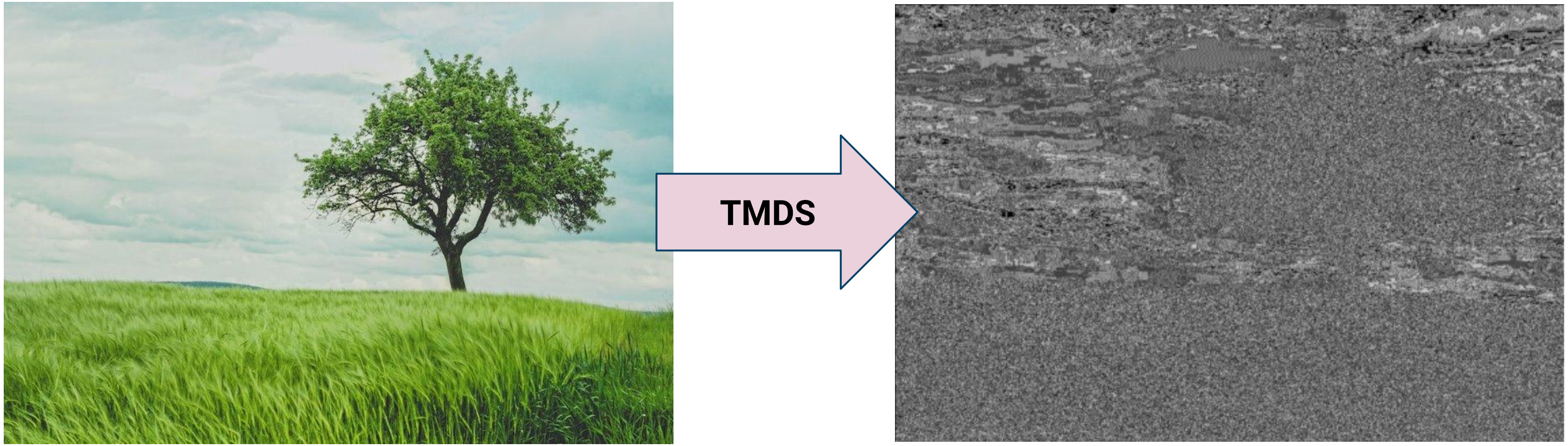
In this section, we introduce the proposed framework. We begin with a discussion on the perceptual processing unit, including encoding, prediction and feature reconstruction. We continue with an explanation of the self-supervised approach for training the model, followed by a discussion on boundary detection and adaptive learning. We conclude with implementation details of the proposed approach. It is to be noted that [25] also propose a similar approach based



HDMI Digital signal
Transition Minimized Differential Signaling

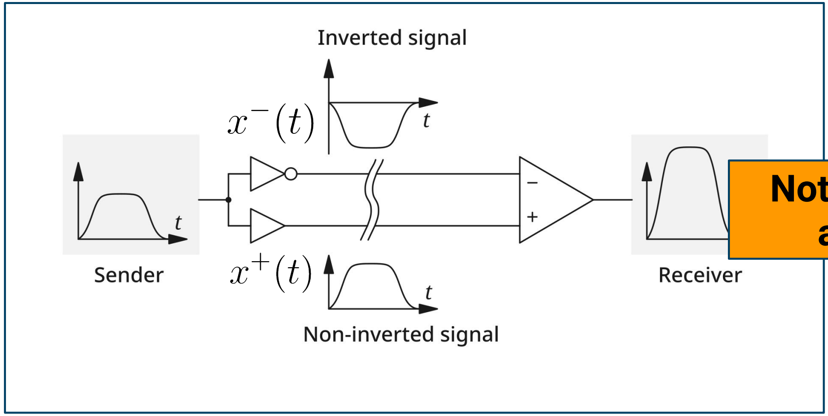
Each color intensity (0-255) represented with 8 bits

➔ Mapped to a 10 bit word (non-linear and with memory)

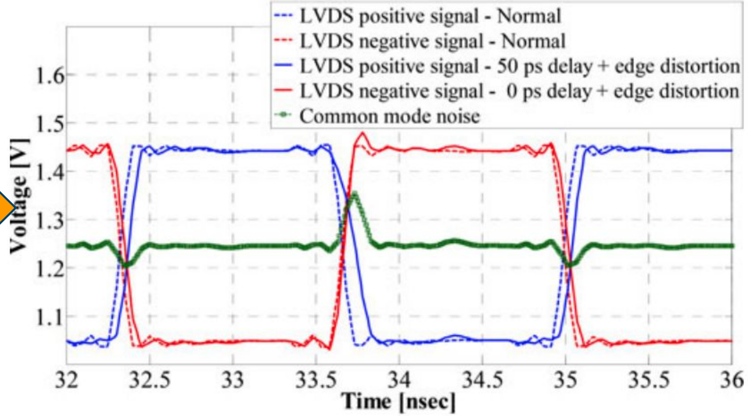


HDMI Electrical signal
Transition Minimized **Differential Signaling**

Each bit is transmitted as a differential pair:



Not perfectly aligned



$$x^+(t) = V_{cc} + \sum_k x_b[k]p(t - kT_b)$$

$$x^-(t) = V_{cc} - \sum_k x_b[k]p(t - kT_b)$$

$$\Rightarrow x(t) = x^+(t) + x^-(t) = 2V_{cc}$$

$$x(t) = 2V_{cc} + \sum_k x_b[k]q(t - kT_b)$$

with $q(t) = p(t) - p(t - \epsilon T_b)$

Tae-Lim Song et al., “Modeling of Leaked Digital Video Signal and Information Recovery Rate as a Function of SNR”, IEEE TEMC, 2014

HDMI Electromagnetic signal

Expression of the signal "seen outside" the HDMI cable (**still a PAM**):

$$x(t) = 2V_{cc} + \sum_k x_b[k]q(t - kT_b)$$

What's the power spectrum? (ignore the offset)

$$S_X(f) = \frac{|Q(f)|^2}{T_b} S_{X_b}(f)$$

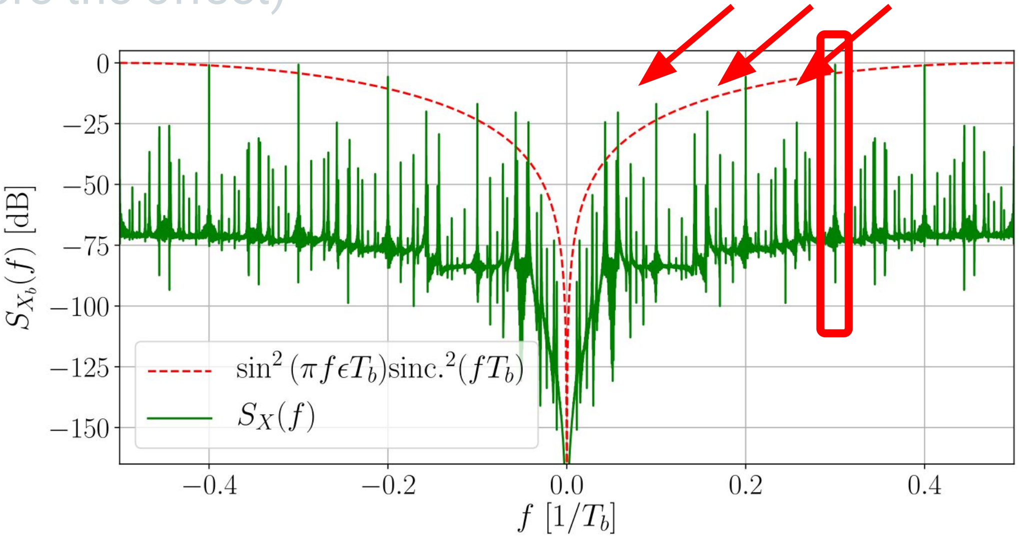
PSD of the encoded bits?
Let's look at a simulation…

What about Q(f)?

$$S_{X_b}(f) = \sum_l R_{X_b}[l]e^{-j2\pi f l T_b}$$

$$R_{X_b}[l] = \mathbb{E}\{x_b[k]x_b[k+l]\}$$

Bits 1 pixel apart are typically the opposite



HDMI Electromagnetic signal

Expression of the signal “seen outside” the HDMI cable (**still a PAM**):

$$x(t) = 2V_{cc} + \sum_k x_b[k]q(t - kT_b)$$

What’s the power spectrum? (ignore the offset)

$$S_X(f) = \frac{|Q(f)|^2}{T_b} S_{X_b}(f)$$

PSD of the encoded bits?
Let’s look at a simulation…

What about Q(f)?

$$S_{X_b}(f) = \sum_l R_{X_b}[l]e^{-j2\pi f l T_b}$$

$$R_{X_b}[l] = \mathbb{E}\{x_b[k]x_b[k+l]\}$$

Bits 1 line apart are typically the same

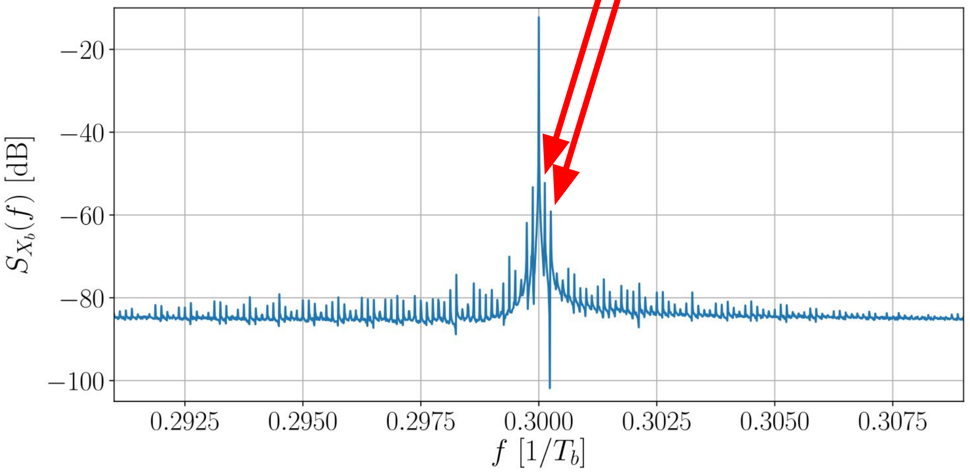
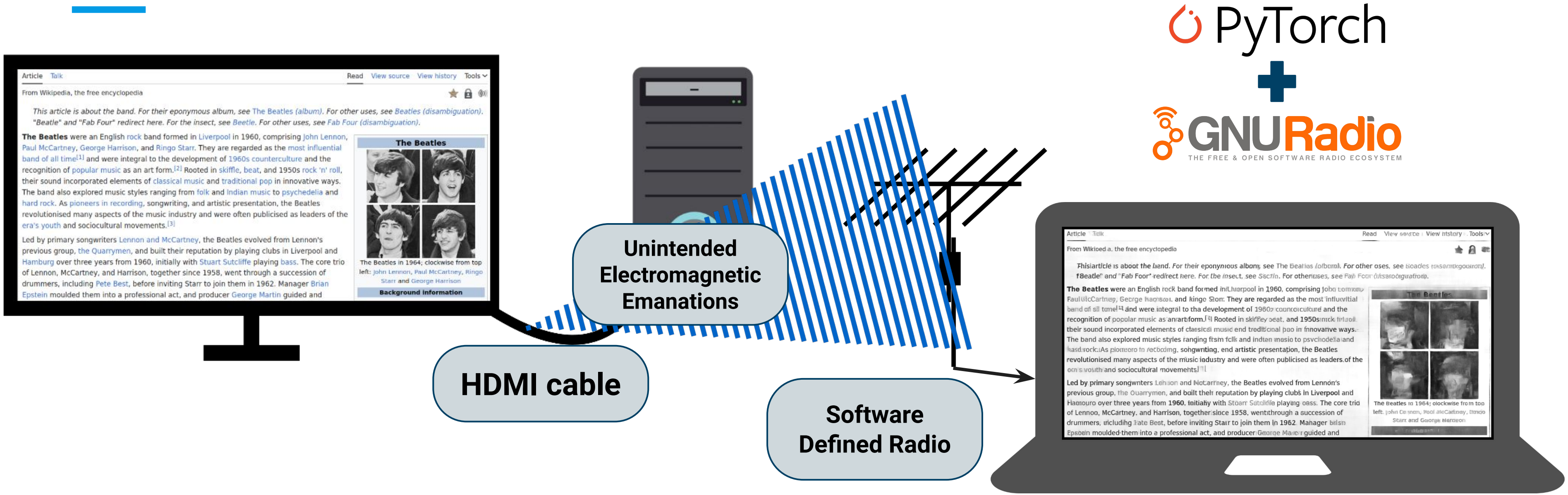


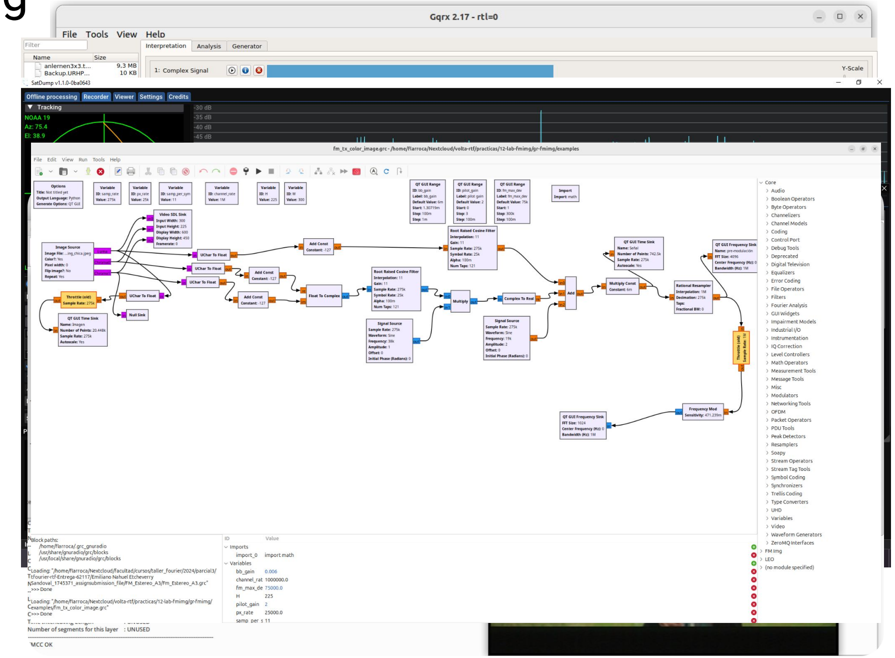
Image recovery roadmap: SDR



Receiving the signal: Software Defined Radio

- Generic hardware <-> Software processing
  - Spectrum analyzer
  - Signal reverse engineering
  - Satellite imagery
  - Digital TV transceiver
  - Custom signal processing chain through

GNURadio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM



GQRX - https://www.gqrx.dk/

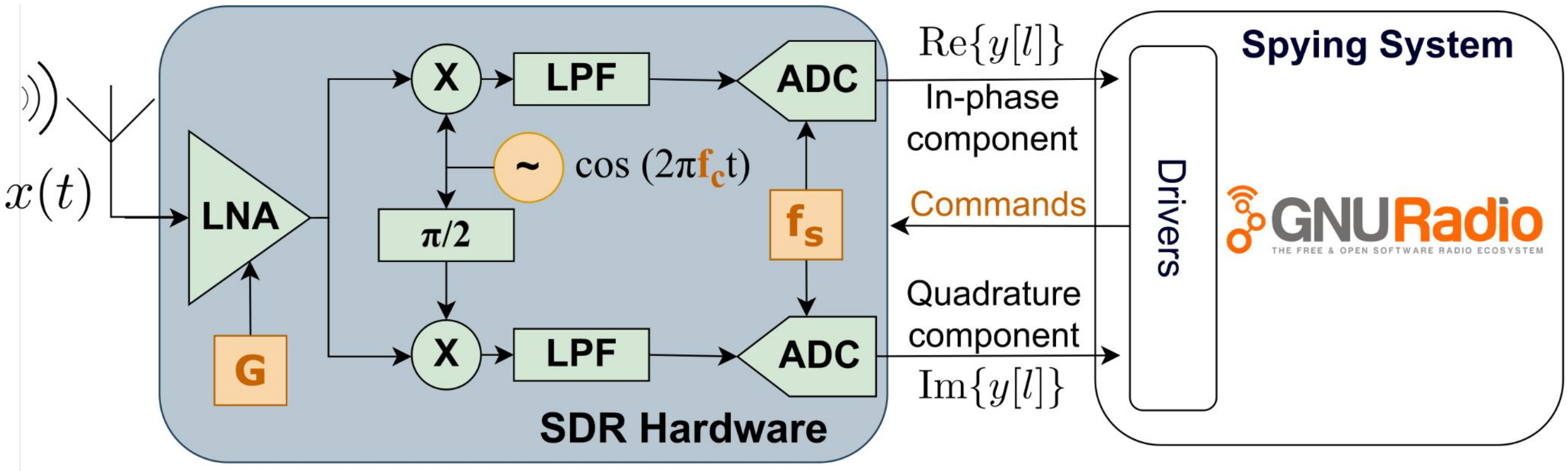
Universal Radio Hacker - https://github.com/jopohl/urh

SatDump - https://www.satdump.org/

gr-isdbt - https://github.com/git-artes/gr-isdbt

Receiving the signal: Software Defined Radio

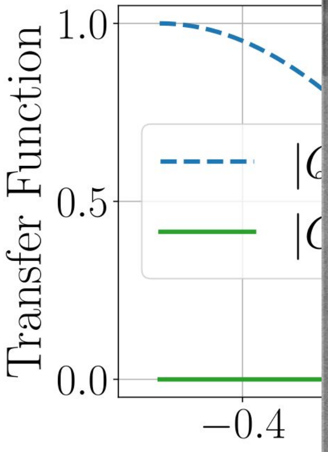
How do we receive the signal?

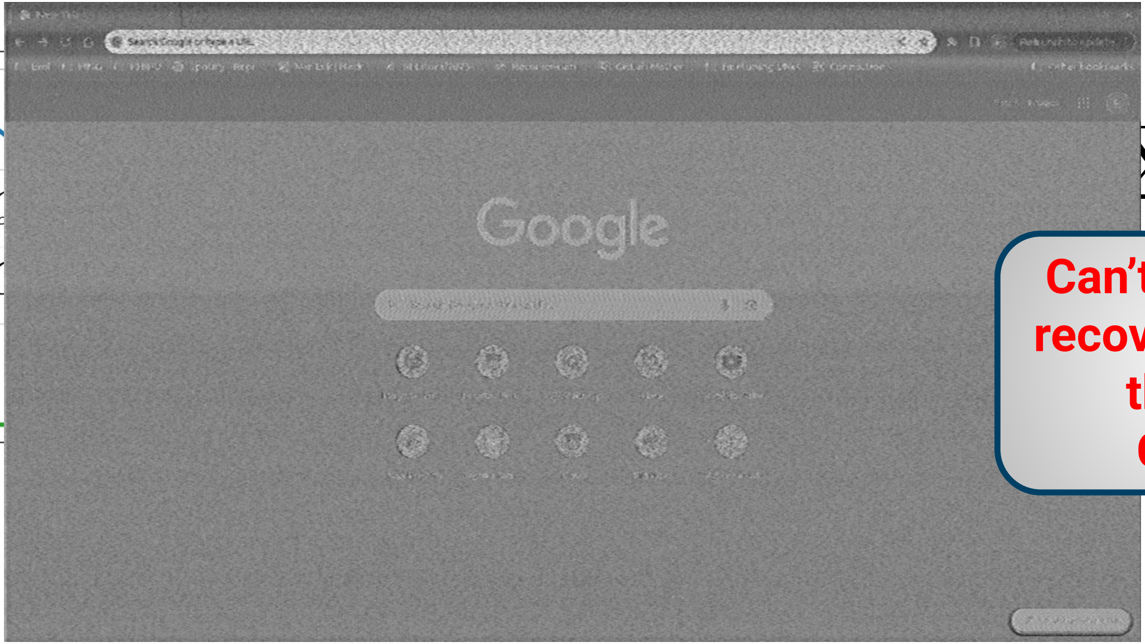


$$x(t) = 2V_{cc} + \sum_k x_b[k]q(t - kT_b)$$

Receiving the signal: Software Defined Radio

Problem: sampling rate fs is much smaller than bit rate





$$\sum_k x_b[k]q(t - kT_b)$$

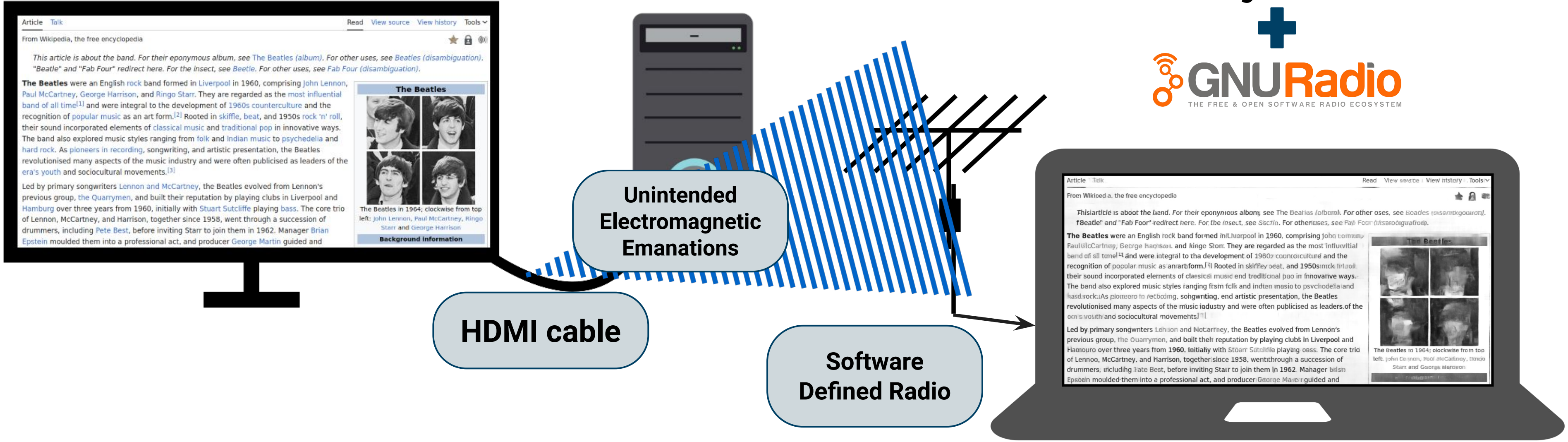
$T_b)$

Can't have better recovering quality than this. Can we?

Image recovery roadmap: deep learning

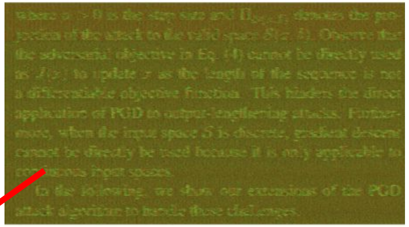
PyTorch

GNURadio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

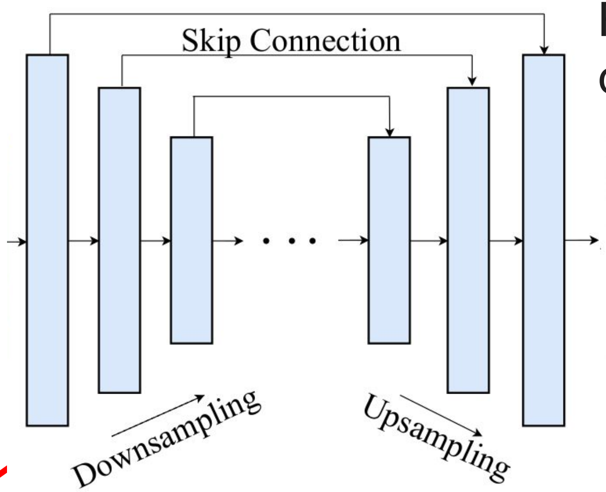


Inverse problem

Given an observation: Y

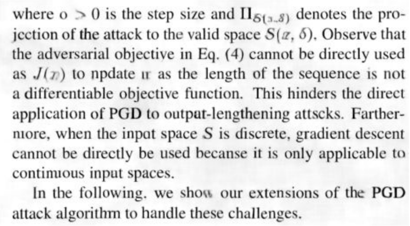


Complex samples aligned by vanilla gr-tempest



DRUNet: deep **CNN architecture**

Find the ***most likely*** original image: $\hat{X}$



where $\alpha > 0$ is the step size and $\Pi_{\mathcal{S}(x,\delta)}$ denotes the projection of the attack to the valid space $\mathcal{S}(x, \delta)$. Observe that the adversarial objective in Eq. (4) cannot be directly used as $J(x)$ to update x as the length of the sequence is not a differentiable objective function. This hinders the direct application of PGD to output-lengthening attacks. Furthermore, when the input space $\mathcal{S}$ is discrete, gradient descent cannot be directly be used because it is only applicable to continuous input spaces.

In the following, we show our extensions of the PGD attack algorithm to handle these challenges.

Model:

$$Y = \mathcal{T}(X) + N,$$

observation

noise

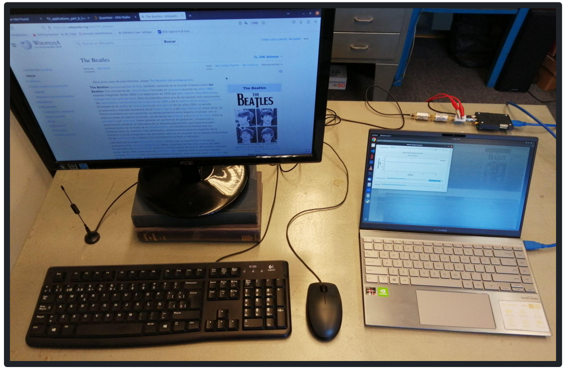
forward operator (TMDS, SDR, etc.)

original image

Kai Zhang et al.,"Plug-and-play image restoration with deep denoiser prior" IEEE TPAMI 2021

Dataset construction $\{X_i, Y_i\}_N$

- **Python + GNU Radio** scripts for synthetic data **(2189 images)**
  - “Free” to produce at any computer/server
  - No need of experimental setup
  - ***Low-cost generation***
- ***Real-life*** image observations! **(1302 images)**
  - **Protocol** with experimental setup (necessary for ***supervised learning*** approach)
  - Took around **65 hs** to acquire
  - ***High-cost generation and time consuming***



- A total of **N=3491** images pairs for *training/validation/test*

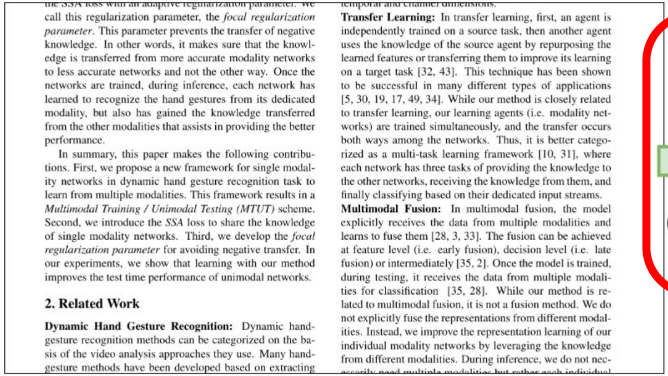
Results!

Original

Vanilla tempest
deep-tempest

Evaluation metric (*text readability*)

Original



the SSA loss with an adaptive regularization parameter. We call this regularization parameter, the *focal regularization parameter*. This parameter prevents the transfer of negative knowledge. In other words, it makes sure that the knowledge is transferred from more accurate modality networks to less accurate networks and not the other way. Once the networks are trained, during inference, each network has learned to recognize the hand gestures from its dedicated modality, but also has gained the knowledge transferred from the other modalities that assists in providing the better performance.

In summary, this paper makes the following contributions. First, we propose a new framework for single modality networks in dynamic hand gesture recognition task to learn from multiple modalities. This framework results in a *Multimodal Training / Unimodal Testing (MTUT)* scheme. Second, we introduce the *SSA* loss to share the knowledge of single modality networks. Third, we develop the *focal regularization parameter* for avoiding negative transfer. In our experiments, we show that learning with our method improves the test time performance of unimodal networks.

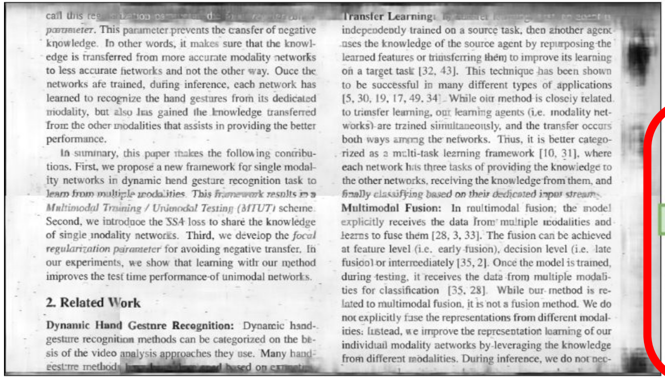
2. Related Work

Dynamic Hand Gesture Recognition: Dynamic hand-gesture recognition methods can be categorized on the basis of the video analysis approaches they use. Many hand-gesture methods have been developed based on extracting

temporal and channel dimensions.

Transfer Learning: In transfer learning, first, an agent is independently trained on a source task, then another agent uses the knowledge of the source agent by repurposing the learned features or transferring them to improve its learning on a target task [32, 43]. This technique has been shown to be successful in many different types of applications [5, 30, 19, 17, 49, 34]. While our method is closely related to transfer learning, our learning agents (i.e. modality networks) are trained simultaneously, and the transfer occurs both ways among the networks. Thus, it is better categorized as a multi-task learning framework [10, 31], where each network has three tasks of providing the knowledge to the other networks, receiving the knowledge from them, and finally classifying based on their dedicated input streams.

Multimodal Fusion: In multimodal fusion, the model explicitly receives the data from multiple modalities and learns to fuse them [28, 3, 33]. The fusion can be achieved at feature level (i.e. early fusion), decision level (i.e. late fusion) or intermediately [35, 2]. Once the model is trained, during testing, it receives the data from multiple modalities for classification [35, 28]. While our method is related to multimodal fusion, it is not a fusion method. We do not explicitly fuse the representations from different modalities. Instead, we improve the representation learning of our individual modality networks by leveraging the knowledge from different modalities. During inference, we do not necessarily need multiple modalities but rather each individual



parameter. This parameter prevents the transfer of negative knowledge. In other words, it makes sure that the knowledge is transferred from more accurate modality networks to less accurate networks and not the other way. Once the networks are trained, during inference, each network has learned to recognize the hand gestures from its dedicated modality, but also has gained the knowledge transferred from the other modalities that assists in providing the better performance.

In summary, this paper makes the following contributions. First, we propose a new framework for single modality networks in dynamic hend gesture recognition task to learn from multiple modalities. This framework results in a *Multimodal Training / Unimodal Testing (MTUT)* scheme. Second, we introduce the *SSA* loss to share the knowledge of single modality networks. Third, we develop the *focal regularization parameter* for avoiding negative transfer. In our experiments, we show that learning with our method improves the test time performance of unimodal networks.

2. Related Work

Dynamic Hand Gesture Recognition: Dynarcic hand-gesture recognition methods can be categorized on the basis of the video analysis approaches they use. Many hand-

Transfer Learning: independently trained on a source task, then another agent uses the knowledge of the source agent by repurposing the learned features or transferring them to improve its learning on a target task [32, 43]. This technique has been shown to be successful in many different types of applications [5, 30, 19, 17, 49, 34]. While our method is closely related to transfer learning, our learning agents (i.e. modality networks) are trained simultaneously, and the transfer occurs both ways among the networks. Thus, it is better categorized as a multi-task learning framework [10, 31], where each network has three tasks of providing the knowledge to the other networks, receiving the knowledge from them, and finally classifying based on their dedicated input streams.

Multimodal Fusion: In multimodal fusion, the model explicitly receives the data from multiple modalities and learns to fuse them [28, 3, 33]. The fusion can be achieved at feature level (i.e. early fusion), decision level (i.e. late fusion) or intermediately [35, 2]. Once the model is trained, during testing, it receives the data from multiple modalities for classification [35, 28]. While our method is related to multimodal fusion, it is not a fusion method. We do not explicitly fuse the representations from different modalities. Instead, we improve the representation learning of our individual modality networks by leveraging the knowledge from different modalities. During inference, we do not nec-

Reconstruction

OCR

In summary, this paper makes the following contributions. First, we propose a new framework for single modality networks in dynamic hand gesture recognition task to learn from multiple modalities. This framework results in a Multimodal Training / Unimodal Testing (MTUT) scheme.

2. Related Work

Dynamic Hand Gesture Recognition: Dynamic hand-gesture recognition methods can be categorized on the basis of the video analysis approaches they use. Many hand-
...

CER=14.5%

Character Error Rate

OCR

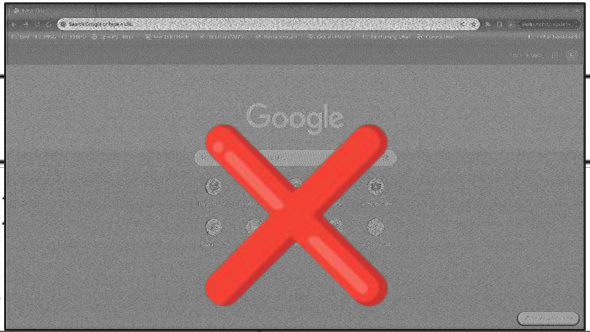
if suminary, this paper rhakes the following contributions. First, we proposé a new framework for single modality networks in dynamic hend gesture recognition task to leam from multiple wodalities. This framework resnits im 9 Mulitio1odal Training / Urinodal Testing (21TUT) scheme-

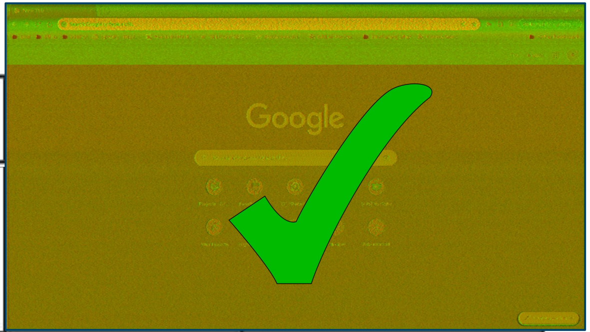
2. Related Work

Dynamic Hand Gesture Recognition: Dynarcic hand--gesture recognition methods can be categorized on the be-sis of the video analysis approaches they use. Many hand-
...

Experiments:

- Training with real-life data only (*pure model*):





Model			
Raw image mag (gr-tempes			
Pure (w/ complex values)	**15.2**	**0.787**	**35.3**
Pure (w/ magnitude only)	14.2	0.754	43.6

Evaluation with real-life testset

Experiments:

- Train with low-cost synthetic data (*base model*) and finetune with a fraction of real-life dataset:

Fraction	F[illegible]	[illegible]M	CER (%)
5%	[illegible]	[illegible]	39.0
10%	[illegible]	[illegible]	35.0
20%	[illegible]	[illegible]	33.3
50%	[illegible]	[illegible]	31.4
100%	[illegible]	[illegible]	29.8

Synthetic data trained model is a good starting point

Better performance by including more real-life data

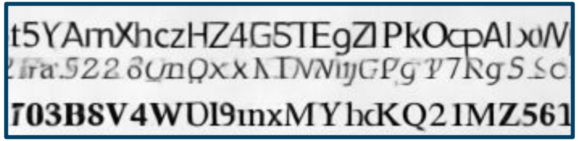
Evaluation with real-life testset

Robustness

Capacity for non-trained conditions

- For *never-seen* fonts
  - Generated images with *random text fonts* and its observations
  - Using our best model → **CER = 48%** for test images 😔
  - Training 10 epochs → drops to **CER = 30%** (same test images) 😊

Original



Best model

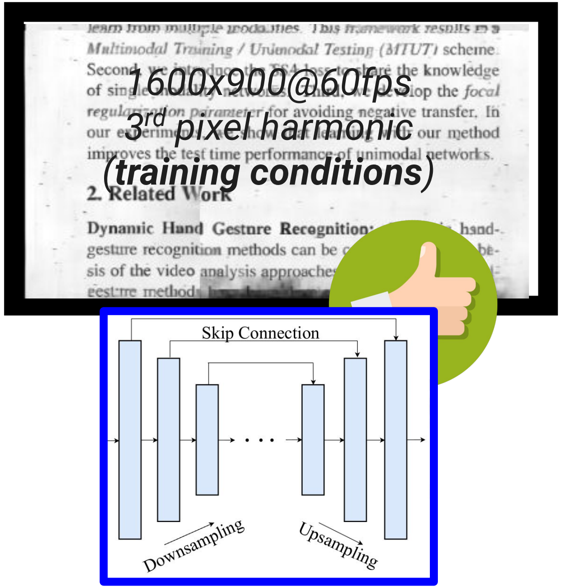
Fine tuned
best model

Robustness

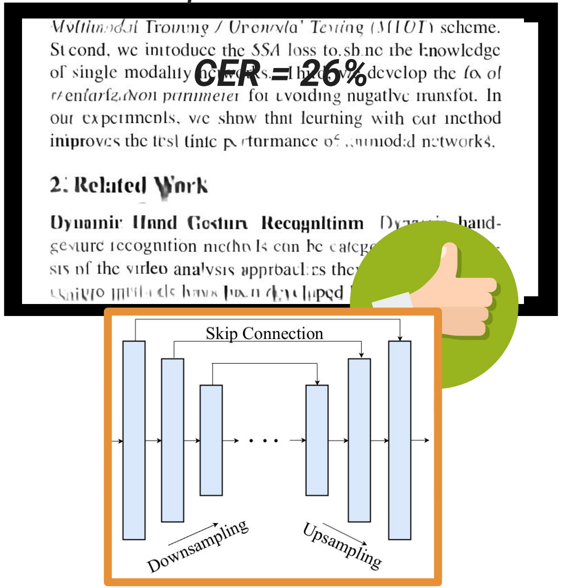
Capacity for non-trained conditions (*cont.*)

- Changing *PC user setup*

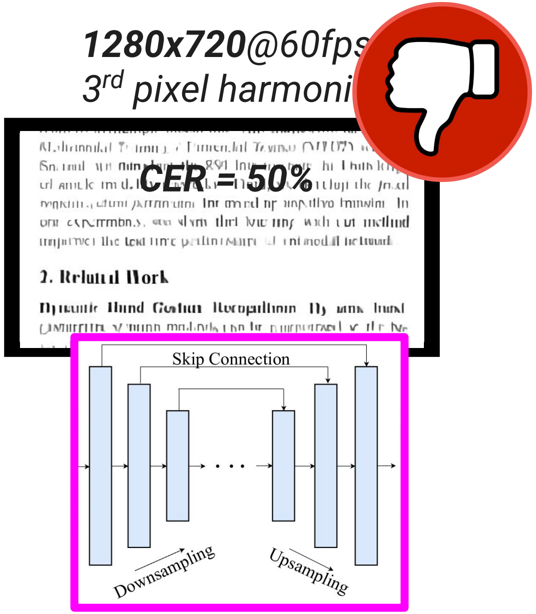
Synthetic data generator!
One *base model* per resolution & harmonic *configuration*



Same resolution
4^{rd} *pixel harmonic*



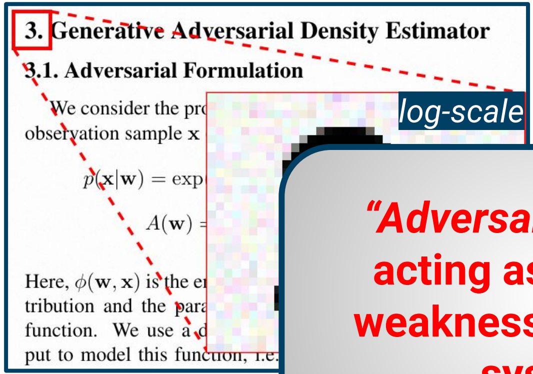
1280x720@60fps
3^{rd} pixel harmoni

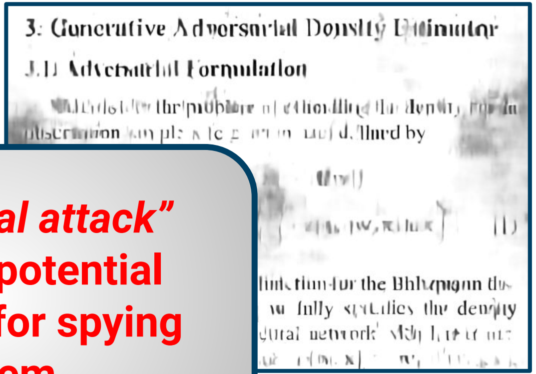


Countermeasures

Imperceptible changes at pixel values → high observation variance

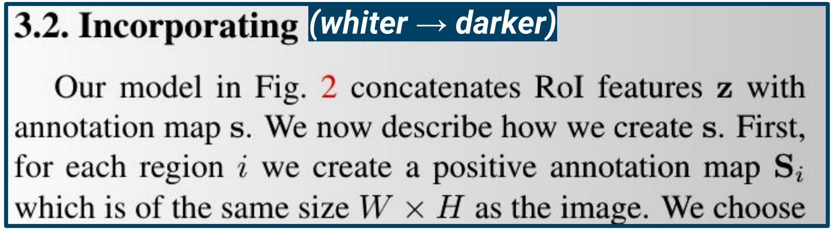
- Add low level noise to frames of display monitor





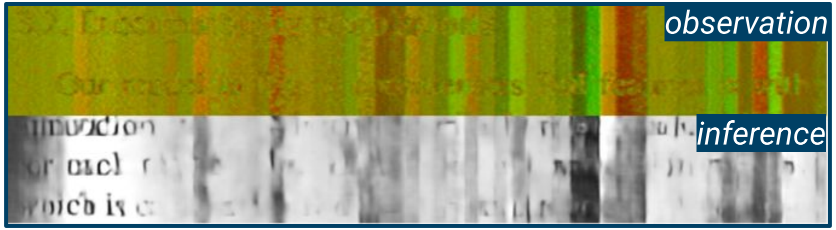
“Adversarial attack” acting as potential weakness for spying system

- Color gradient background



3.2. Incorporating

Our model in Fig. 2 concatenates RoI features $\mathbf{z}$ with annotation map $\mathbf{s}$. We now describe how we create $\mathbf{s}$. First, for each region i we create a positive annotation map $\mathbf{S}_i$ which is of the same size $W \times H$ as the image. We choose



Available Resources

- Code for synthesis of image observation for involuntary electromagnetic emanations for HDMI cables.

- A total of 1302 real-life samples for train/test.

- *deep-tempest* GNU-Radio framework (*gr-tempest2.0* + trained model).

All available (*open-source*) at github.com/emidan19/deep-tempest

Awareness ✅
We are on the news

RTL-SDR.COM

RTL-SDR (RTL2832U) and software defined radio news and projects. Also featuring Airspy, HackRF, FCD, SDRplay and more.

HOME | ABOUT RTL-SDR | QUICK START GUIDE | FEATURED ARTICLES | SOFTWARE | SIGNAL ID WIKI | FORUM | RTL-SDR STORE

JULY 24, 2024

DEEP-TEMPEST: EAVESDROPPING ON HDMI VIA SDR AND DEEP

are-defined radio. TEMPEST aka (Van
d are able to recover information from
at allows you to view images from a

NewScientist

Technology

AI can reveal what's
via signals leaking fr

Electromagnetic radiation leaking from the cab
can be intercepted and decoded by AI to reveal

By Ma

2

It's TOO Easy to Accidentally Do Illegal Stuff with This

hackster.io
AN AVNET COMMUNITY

Search

Projects | Channels | News

Deep-TEMPEST Reveals All

Deep-TEMPEST is an exploit that leverages an SDR receiver and deep learning to wirelessly reveal what is being displayed on an HDMI monitor.

PCWorld

NEWS | BEST PICKS | REVIEWS | HOW-TO | DEALS

NEWS

Hackers can wirelessly watch your display via HDMI radiation

A newly discovered technique combines wireless EM monitoring and AI algorithms to "read" text on a victim's screen via HDMI radiation, and it's already being used in the wild.

Conclusions

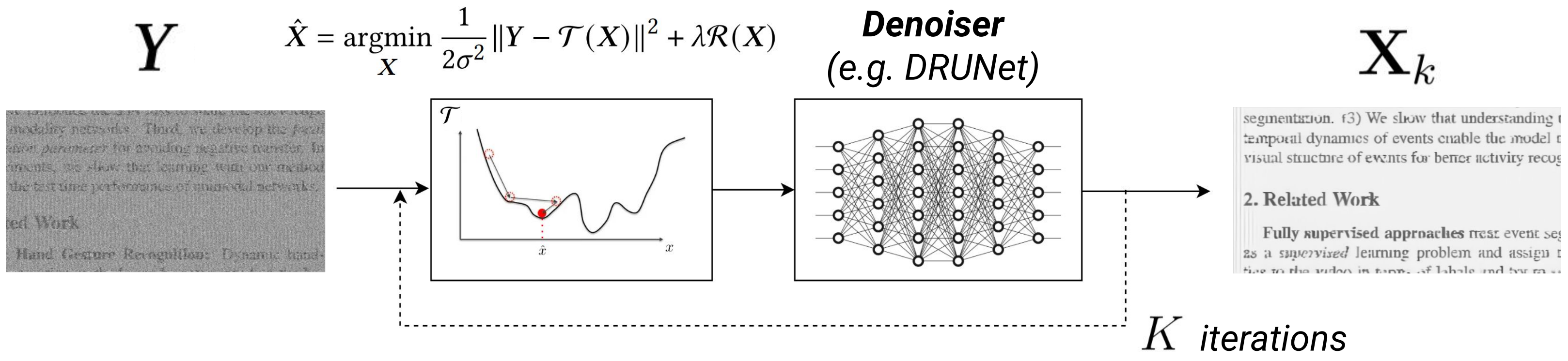
What we've done?

- **Open-source** implementation of mapping operator from a HDMI electromagnetic signal emission observation to its original image.
- **Mathematical formulation** and open-source code for forward operator.
- **Significantly better results** than previous implementations.
- Found **countermeasures** for our system.

Conclusions

What's next?

- ***Plug & Play*** methods, **exploit analytical expression** of forward operator ***instead*** *of retrain* ***one model per user configuration***.



- Use ***time redundancy of video frames*** for better infer quality.

Thank you!

💻🔑🛡️

Questions?

backup slides

For hardware geeks 🛠️⚙️💻

RF:

- Monopole antenna
- Mini-Circuits: ZJL-6G+ amplifier, SLP-450+ LPF and SHP-250+ HPF
- Ettus Research USRP 200-mini

Spying PC:

- AMD Ryzen 5000
- 8 GB RAM

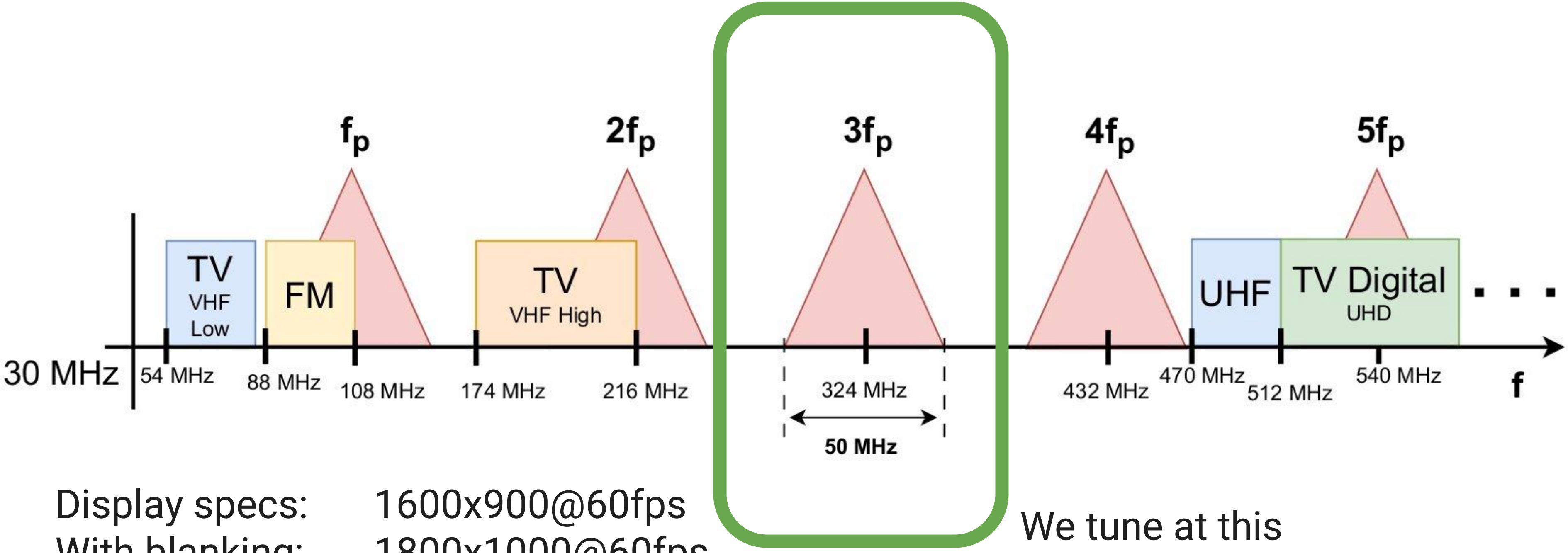
Training PC:

- Intel Core i7-10700F
- 64 GB RAM
- NVIDIA GeForce RTX 3090 24 GB of VRAM

For machine learning geeks 👨🏾‍💻🗽📉

- Pytorch (*of course*)
- Patch size of 256 x 256 pixels
- Batch size of 48 patches
- Adam optimizer
- Learning rate of 1.56×10^{-5}
- TV regularization weight 2.2×10^{-13}
- Hyperparameters search with Optuna
- He’s Normal weights initialization
- 32.638.656 learnable parameters (as default as DRUNet's repo)

Pixel harmonic frequency choice

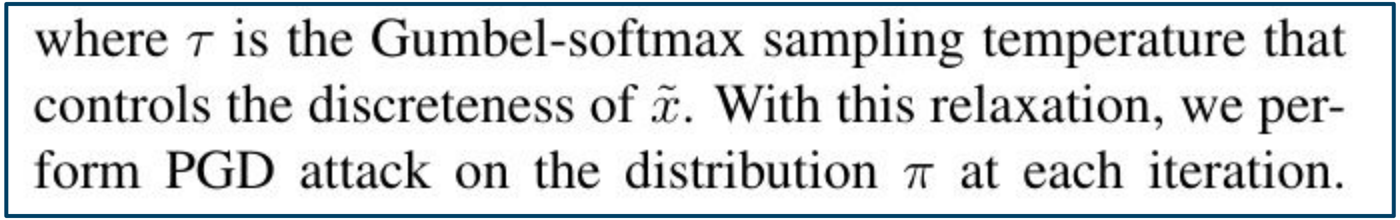


We tune at this pixel harmonic

Display specs:	1600x900@60fps
With blanking:	1800x1000@60fps
Pixel rate:	108 MHz

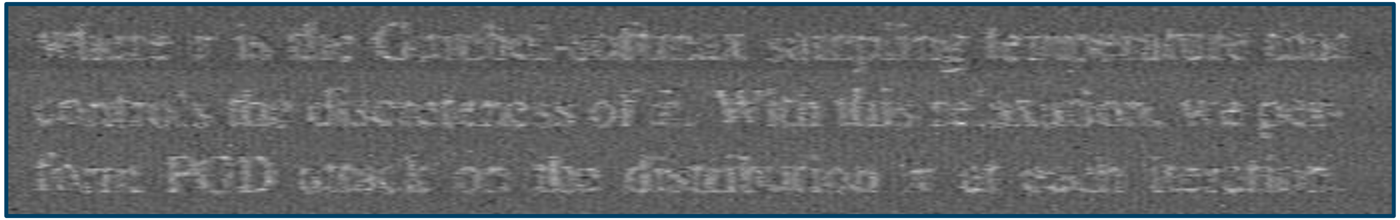
More results

Monitor display view

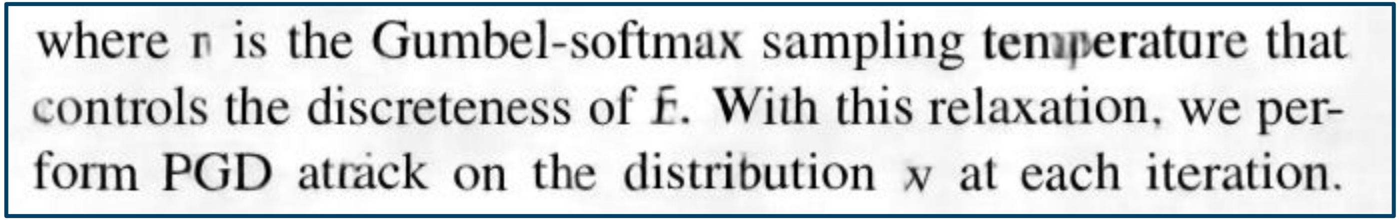


where τ is the Gumbel-softmax sampling temperature that controls the discreteness of $\tilde{x}$. With this relaxation, we perform PGD attack on the distribution π at each iteration.

Vanilla tempest results



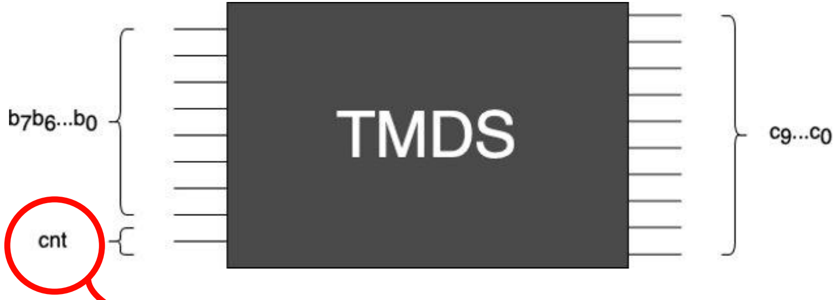
Deep-tempest results



How do we get the signal? *(forward operator 1)*

HDMI uses *Transition Minimized Differential Signaling* (***TMDS***) encoding:

1. 8 bit pixel → 10 bit word (***non linear***)

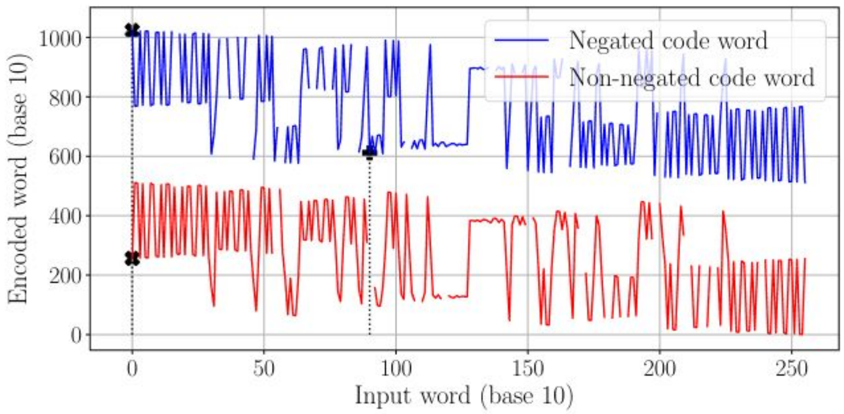


Value *l* of pixel amplitude	TMDS coding	
	First 10-bit word	Second 10-bit word
0	0100000000	1111111111
150	0011011000	1000100111
251	0000000011	1011111100
255	0011111111	1000000000

How do we get the signal?
(forward operator 1)

HDMI uses *Transition Minimized Differential Signaling* (***TMDS***) encoding:

1. 8 bit pixel → 10 bit word (***non linear***)



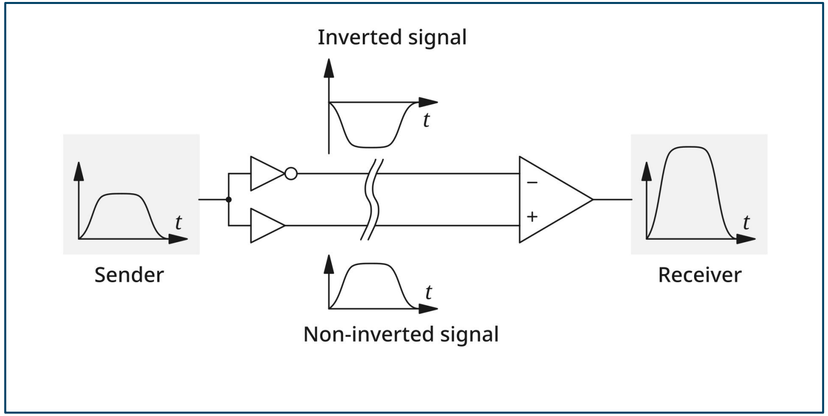
How do we get the signal? *(forward operator 2)*

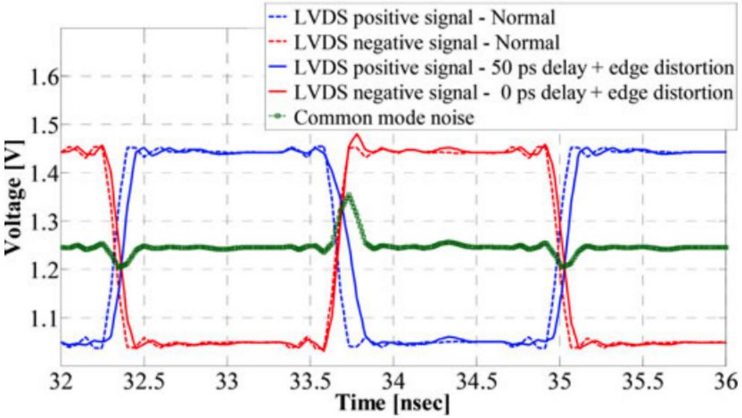
1. Expression of the signal “seen outside” the HDMI cable (***PAM***)

$$x(t) = x^+(t) + x^-(t) = 2V_{cc} + \sum_k x_b[k]\boxed{q(t-kT_b)},$$

where $q(t) = \boxed{p(t)} - \boxed{p(t-\epsilon T_b)}$

$\epsilon = 0.1$ *(really small)*





How do we get the signal?
(forward operator 2)

1. Expression of the signal "seen outside" the HDMI cable (***PAM***)

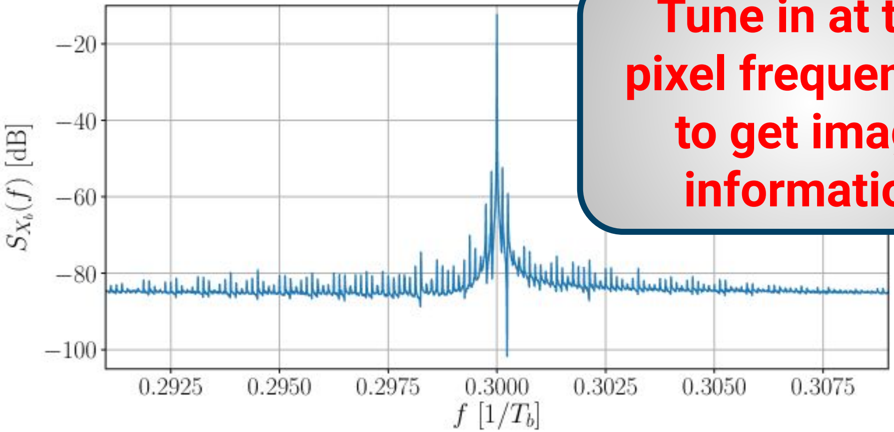
$$x(t) = x^+(t) + x^-(t) = 2V_{cc} + \sum_k x_b[k] \boxed{q(t - kT_b),}$$

where $q(t) = p(t) - p(t - \epsilon T_b)$.

$\epsilon = 0.1$

2. Spectrum of ***$x_b[k]$*** contains information of signal

Spectrum centered at 0.3 f_b = 3 f_p
(3rd *pixel harmonic*)



Tune in at this pixel frequencies to get image information

How do we get the signal?
(forward operator 2)

- Expression of what we get:

$$y[l] = \sum_k x_b[k]g(l/f_s - kT_b).$$

whole system characterization

complex valued
(signal baseband representation
at some pixel harmonic)

SDR's sample rate

- ***g ( )*** *composed by:*
   i. ***q ( )*** (***TMDS***)
   ii. baseband demod (***SDR***)
   iii. LPF (***SDR***)

- Challenges:
   - $\mathbf{f_s \ll 1/T_b = f_b}$ $(\mathbf{f_b \sim 20\, f_s})$
     → SDR gets *"avg"* of samples
     (one pixel per sample)
   - Sampling errors →*"twisted" image recovery*
   - Frequency errors at pixel harmonic (sinusoidal patterns)

How to reconstruct the image *(inverse operator)*

- Given the forward formulation

$$Y = \mathcal{T}(X) + \mathbf{N},$$

observation

forward operator

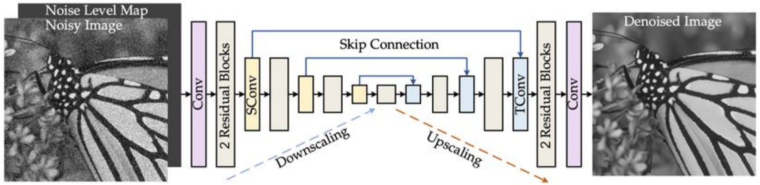
original image

noise

- Find the image that satisfies:

$$\hat{X} = \underset{X}{\operatorname{argmin}} \underbrace{\frac{1}{2\sigma^2}\|Y - \mathcal{T}(X)\|^2}_{\text{data term}} + \underbrace{\lambda \mathcal{R}(X)}_{\text{regularization term } (\textit{total variation})},$$

- **DRUNet** *[Zhang21]* for data term:



Source: "Plug-and-Play Image Restoration with Deep Denoiser Prior"

$$\min_{\Theta} \sum_{i=1}^{N} \mathcal{L}\big(f(Y_i, \Theta), X_i\big).$$

L^2 norm

net (*inverse operator*)

net learnable parameters

TEMPEST / Van Eck Phreaking

- W. Van Eck *[Eck85]* (CRTs)
- M. Kuhn *[Ku03]* (VGA, DVI, HDMI)
- M. Marinov *[Mar14]* (***TempestSDR***)

Source: hackaday.com

UNIVERSITY OF CAMBRIDGE

- P. Menoni *[Men19]* (M.Sc. degree)
- F. Larroca *[Lar22]*, 2020 (***gr-tempest***)
- P. Bertrand *et. al [Bert21]*, 2021 (***gr-tempest2.0***)

Source: gr-tempest2.0

FACULTAD DE INGENIERÍA