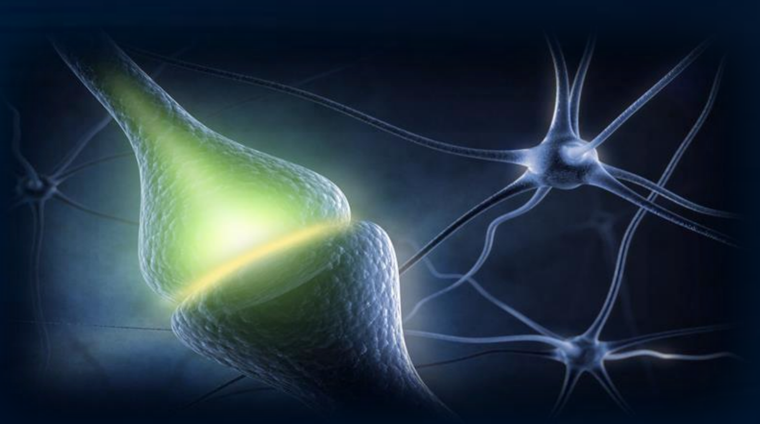
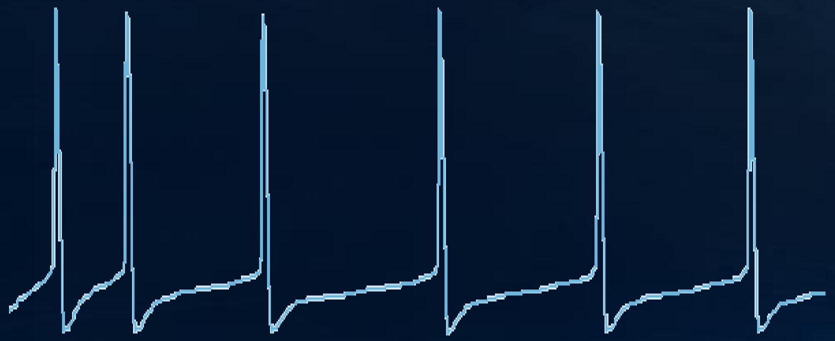




# A RETINA-INSPIRED ENCODER: AN INNOVATIVE STEP ON IMAGE CODING USING LEAKY INTEGRATE-AND-FIRE NEURONS

*Melpomeni DIMOPOULOU, Effrosyni DOUTSI, Marc ANTONINI*

*ICIP 2018*

Université Nice Sophia Antipolis

i3S sophia antipolis

CNRS

UNIVERSITÉ CÔTE D'AZUR

ICIP 2018 – ATHENS, GREECE

2018 IEEE
International Conference
on Image Processing
October 7-10, 2018
Athens, Greece

# Our work:

- The goal: Implementation and experimental testing of a new extended image quantization system which consists of:
  - ➢ the Outer Plexiform Layer (OPL) Filter of the retina [3]
  - ➢ the Leaky Integrate-and-Fire (LIF) Quantizer [4]
- Characteristics of our system:

✓Dynamic

Parametrized by the time

✓Bio-Inspired

Inspired by the mammalian visual system

Encoding is performed by a Leaky Integrate-and-Fire (LIF) neural model

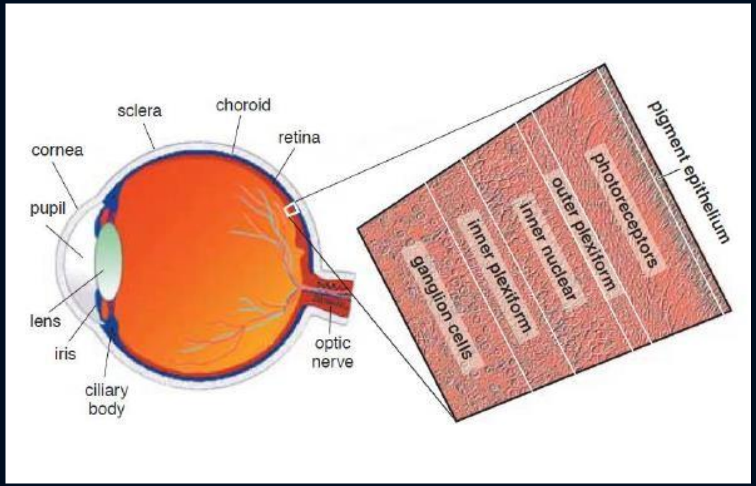
# Why do we need new encoding schemes?

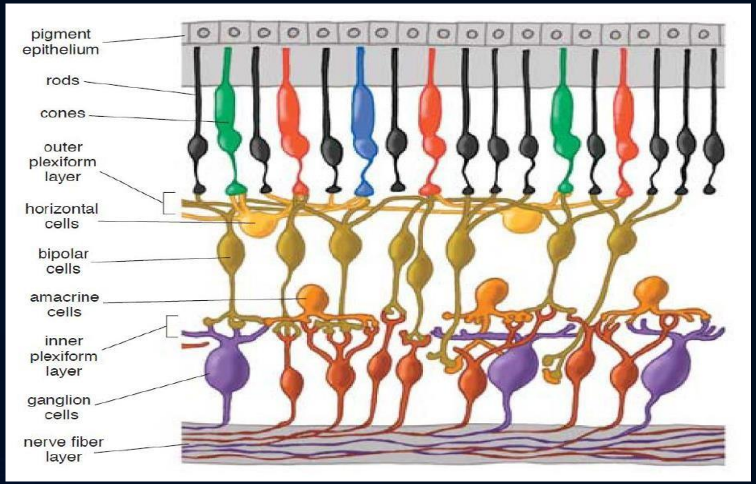
- The amount of media that need to be stored and transmitted increases dramatically
- The latest encoders use similar quantization paradigm and their performance has reached a certain limit.
- There is a high need of finding efficient ways of encoding making use of novel properties.
- Our work suggests a dynamic encoding approach unlike existing schemes that use a static one.

# Our Inspiration – The visual system

- The retina:
  - Outer plexiform layer (OPL)
  - Inner plexiform layer (IPL)
  - Ganglionic layer (GL)

- Ganglion cells:
  - Neurons responsible for the visual data compression reacting to the brightness of light
  - Behave according to the Leaky Integrate-and-Fire (LIF) neural model



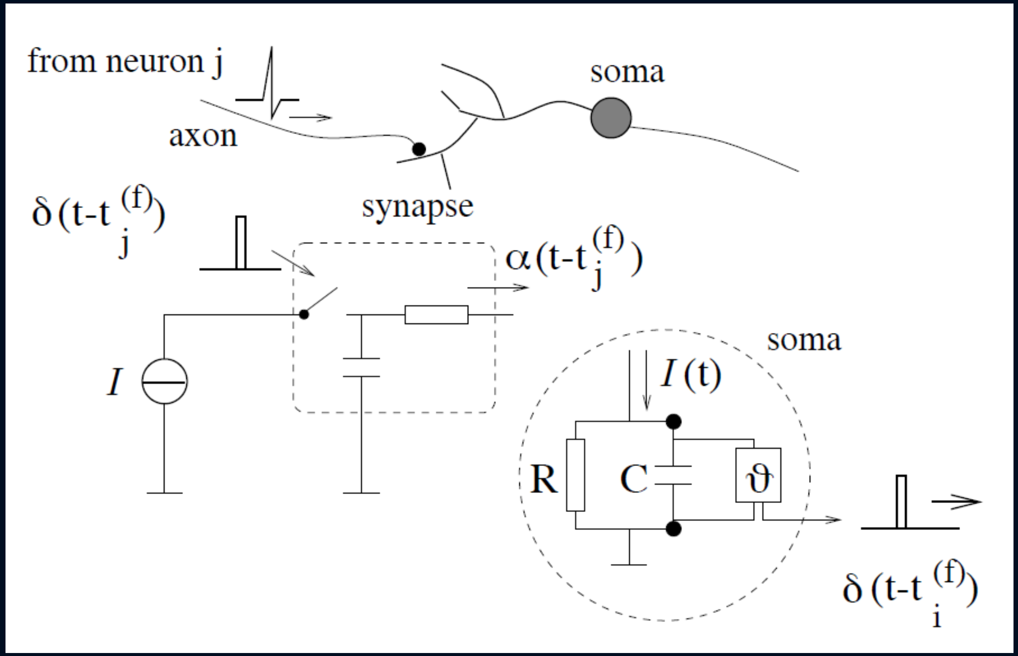



# The LIF model:

- $I(t) = I_R + I_C = \frac{u(t)}{R} + C\frac{du}{dt}$, we set $RC = \tau_m$
- $\tau_m \frac{du}{dt} = -u(t) + RI_0 \Rightarrow$

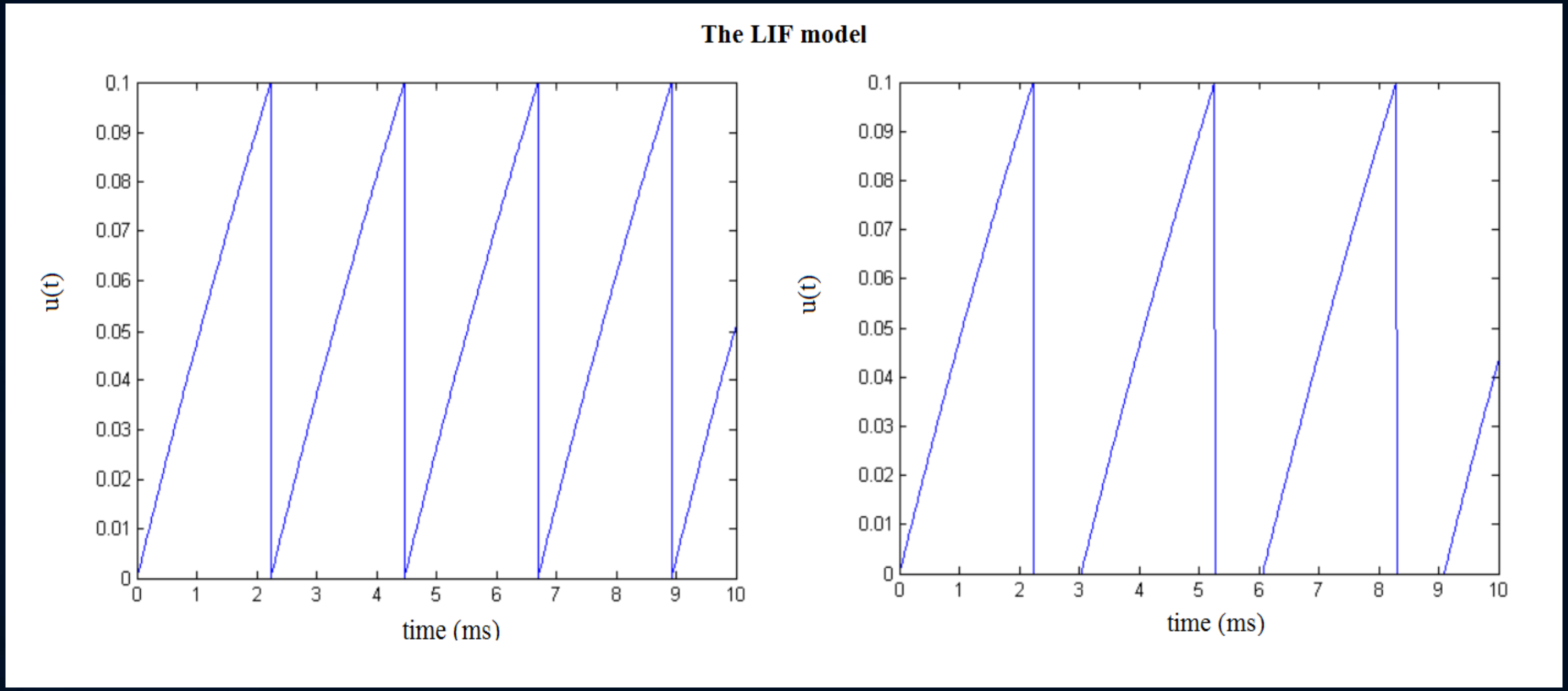
  $$u(t) = RI_0\left(1 - exp\left(-\frac{t-t^k}{\tau_m}\right)\right),$$

  where $t^k$ is the time of a spike occurrence.



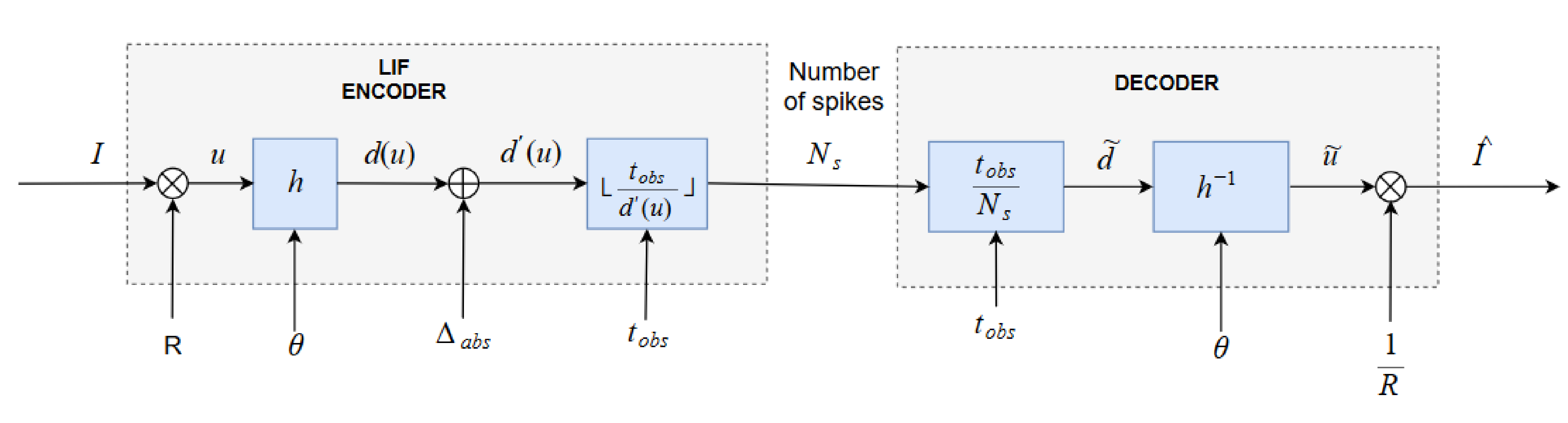

- A spike occurs when: $u(t^{k+1}) = \theta = RI_0\left(1 - exp\left(-\frac{t^{k+1}-t^k}{\tau_m}\right)\right)$

# The LIF behavior




- Computation of the Integration delay : $d(u) = \begin{cases} \infty & , \quad u < \theta \\ h(u;\theta) = \tau_m ln\frac{u}{u-\theta}, & u \geq \theta \end{cases}$
- Computation of the inter-spike delay : $d'(u) = d(u) + \Delta_{abs}$

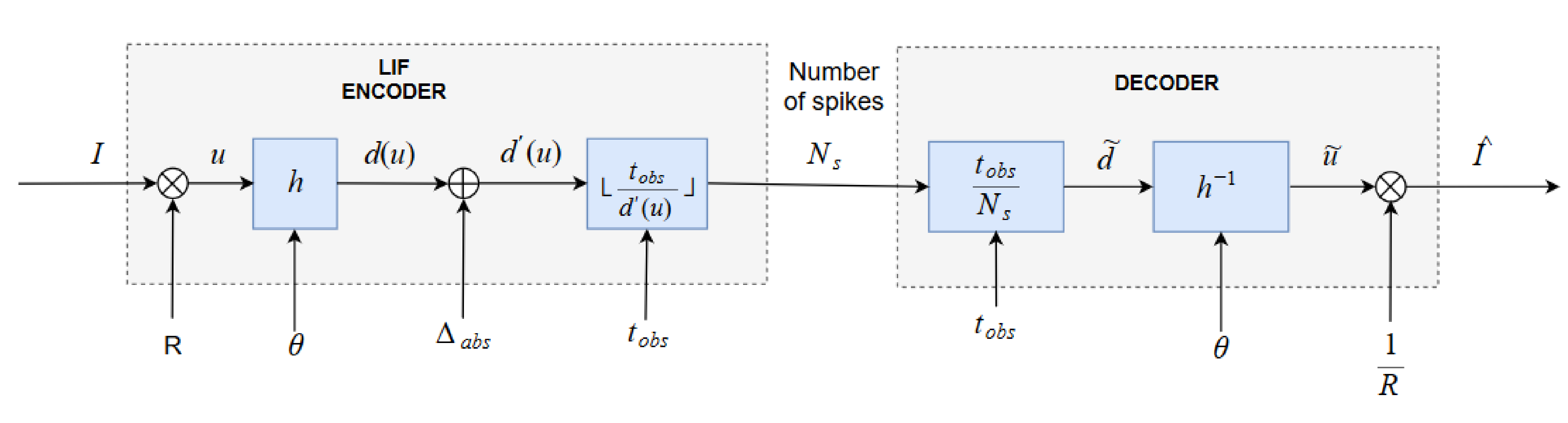
# Our LIF Quantizer – The encoder




- Multiply I with R

Get action potential u

# Our LIF Quantizer – The encoder




- Multiply I with R

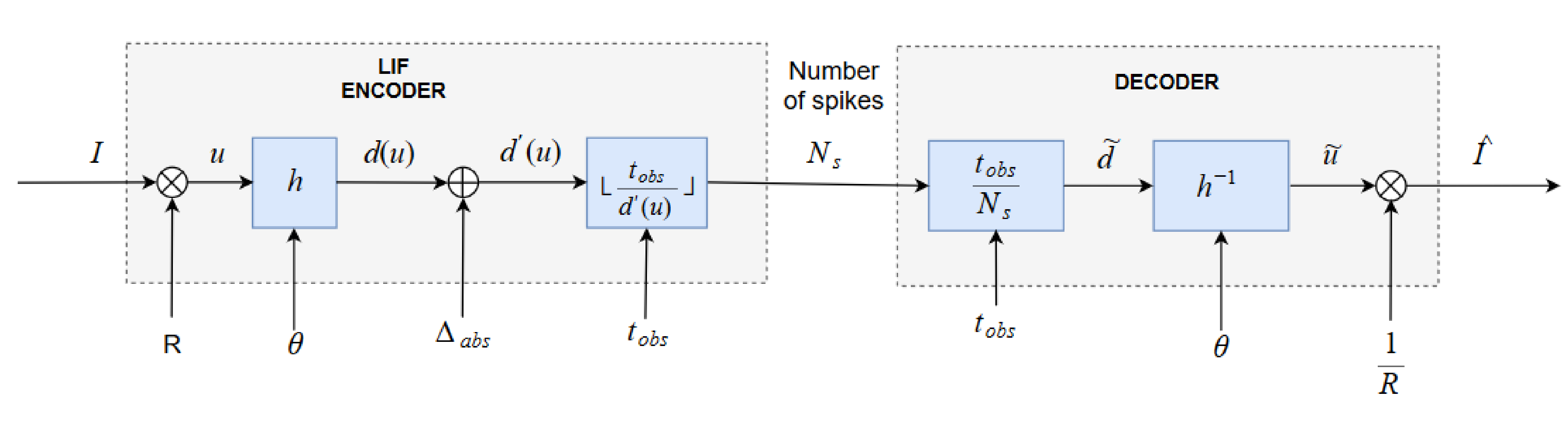
Get action potential u

Compute integration delay

- $d(u) = \begin{cases} \infty & , u < \theta \\ h(u;\theta) = \tau_m ln \frac{u}{u-\theta}, & u \geq \theta \end{cases}$

- Parameters threshold θ, observation time , membrane potential circuit resistance R

# Our LIF Quantizer – The encoder




- Multiply I with R

**Get action potential u**

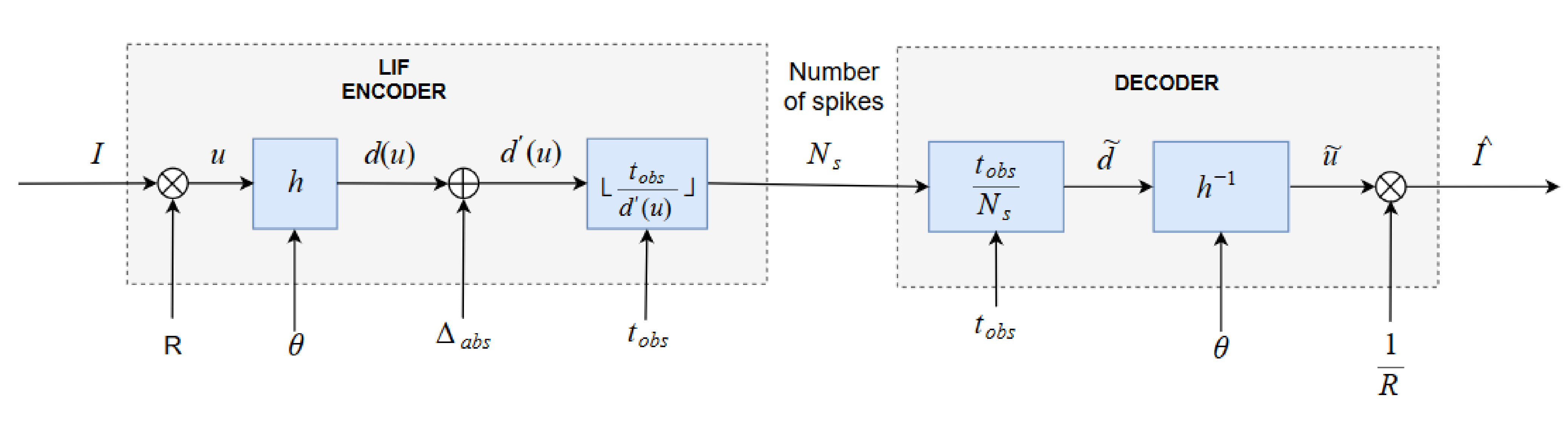
**Compute integration delay**

- $d(u) = \begin{cases} \infty & , u < \theta \\ h(u;\theta) = \tau_m ln\frac{u}{u-\theta}, & u \geq \theta \end{cases}$
- Parameters threshold θ, observation time , membrane potential circuit resistance R

- Add refractory
- $d'(u) = d(u) + \Delta_{abs}$

**Compute spike delay**

# Our LIF Quantizer – The encoder




- Multiply I with R

**Get action potential u**

**Compute integration delay**

- $d(u) = \begin{cases} \infty & , u < \theta \\ h(u;\theta) = \tau_m ln\frac{u}{u-\theta}, & u \geq \theta \end{cases}$
- Parameters threshold θ, observation time , membrane potential circuit resistance R

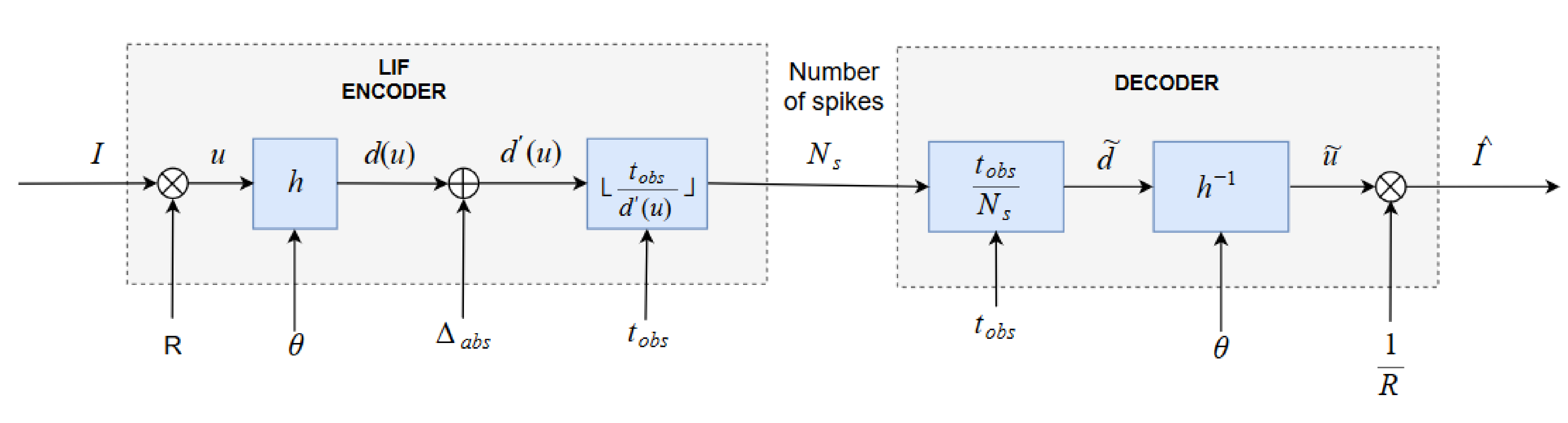
- Add refractory
- $d'(u) = d(u) + \Delta_{abs}$

**Compute spike delay**

**Get number of spikes**

- Divide observation window by the inter-spike delay

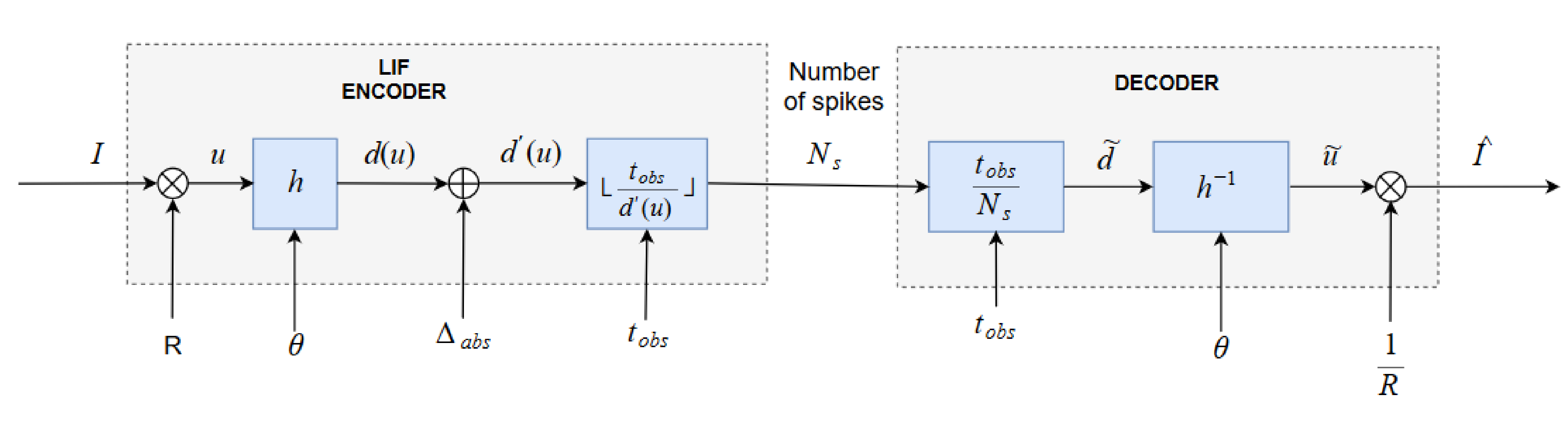
# Our LIF Quantizer - The decoder




- Divide observation window by the number of spikes

Get an approximation of the inter-spike delay

# Our LIF Quantizer - The decoder




Get an estimation of the action potential u

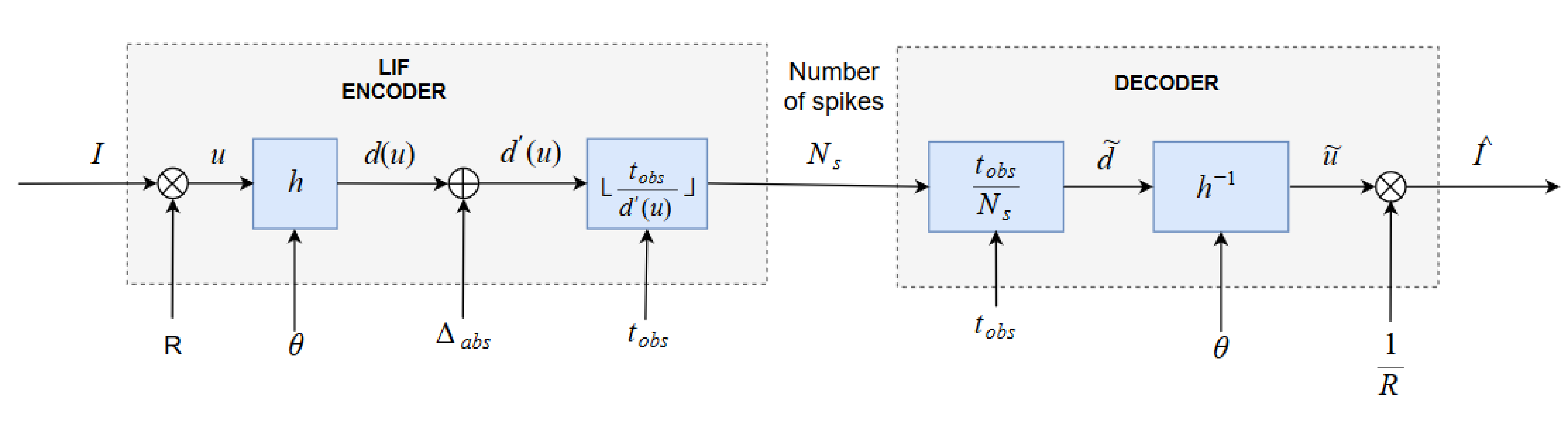
- Use of inverse function of h

- $\tilde{u} = \begin{cases} 0 & , \ \tilde{d}(u) = \infty \\ h^{-1}(\tilde{d}(u);\theta) = \frac{\theta}{1-exp\left(\frac{\tilde{d}(u)}{\tau_m}\right)} & , \ \tilde{d}(u) < \infty \end{cases}$

- Divide observation window by the number of spikes

Get an approximation of the inter-spike delay

# Our LIF Quantizer - The decoder




- Divide observation window by the number of spikes

**Get an approximation of the inter-spike delay**

**Get an estimation of the action potential u**

- Use of inverse function of h

- $\tilde{u} = \begin{cases} 0 & , \tilde{d}(u) = \infty \\ h^{-1}(\tilde{d}(u); \theta) = \frac{\theta}{1-exp\left(\frac{\tilde{d}(u)}{\tau_m}\right)} & , \tilde{d}(u) < \infty \end{cases}$

- Divide estimated u by R

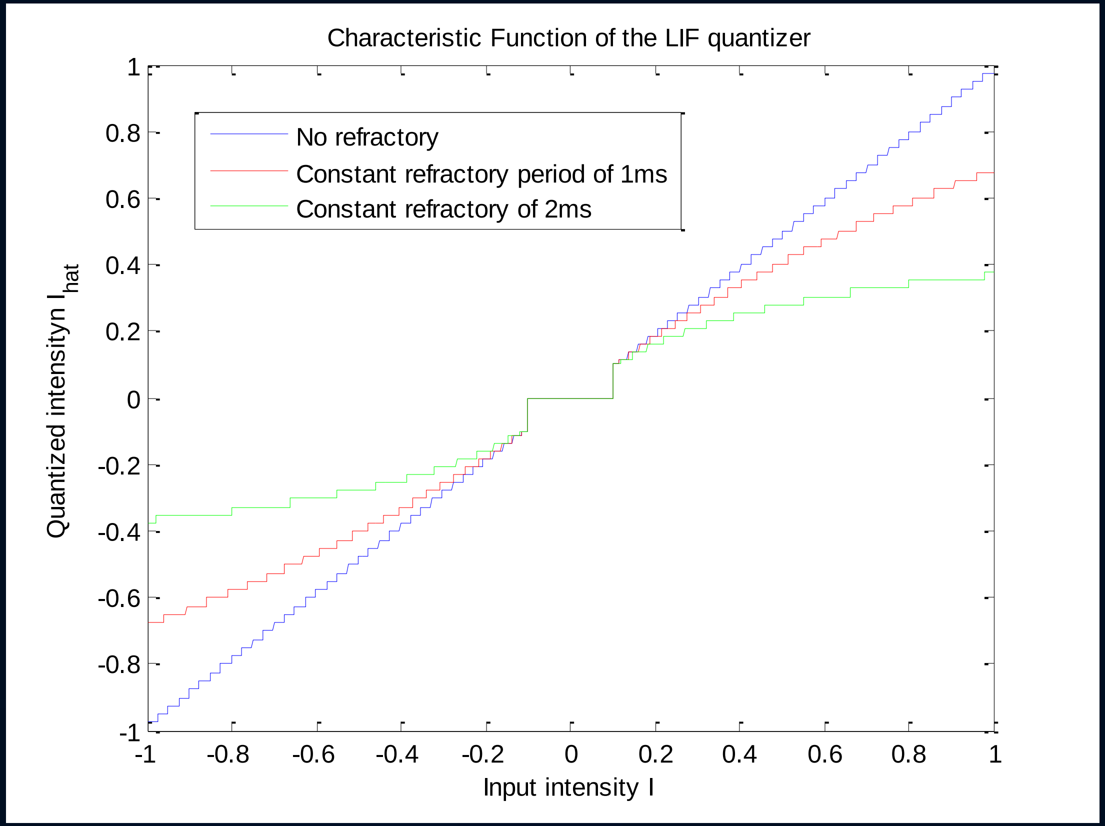
**Get estimation of input intensity**

# The refractory period

- Refractory period works as an additive noise.
- In our experiments the refractory period follows a half Gaussian distribution
- After each spike a positive random refractory period of a specified variance is being generated and added to the inter-spike delay $d(t)$.

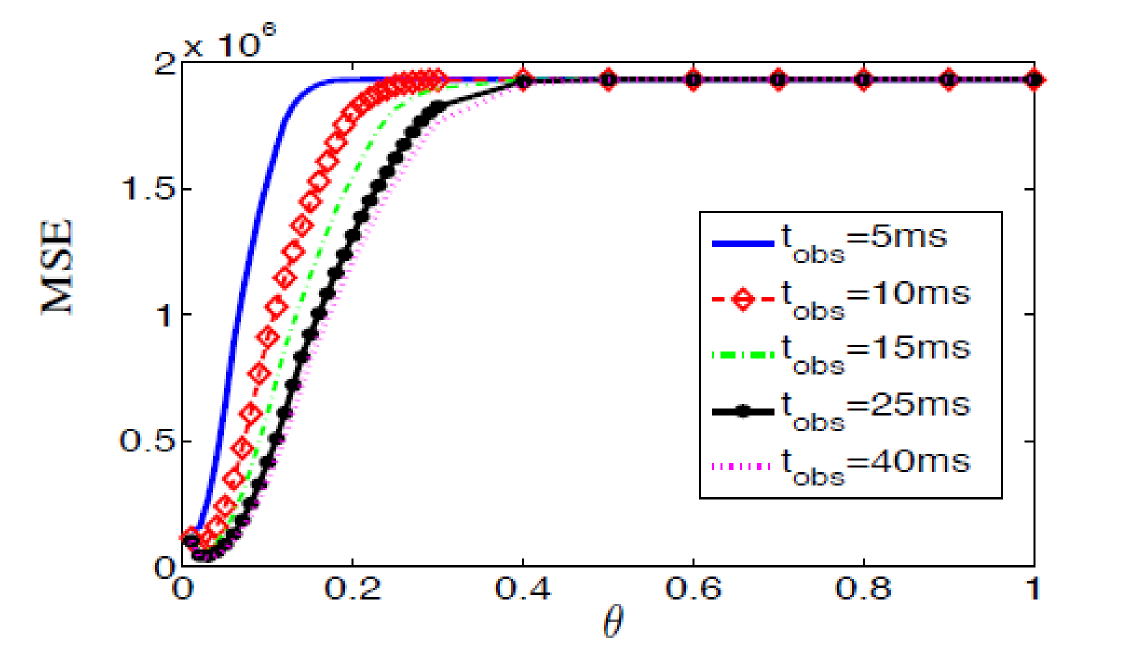
# The LIF characteristic function

- Input intensities are mapped into the estimated quantized intensities computed by our LIF quantizer
- The presence of the refractory period introduces non-linearity



# The MSE in function of the threshold

- The presence of a refractory period introduces overload noise
- This yields the presence of an optimal threshold value which minimizes the MSE.

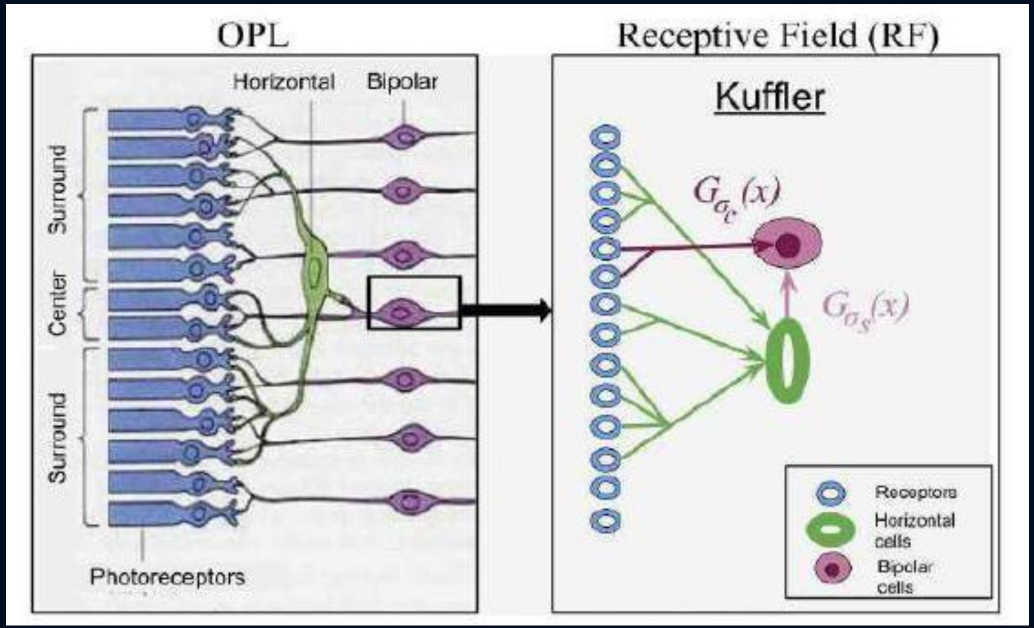


# The OPL filtering

- First layer of the retina
- Receives the visual stimulus $f\ (x, t)$ and produces the equivalent electrical signal using a spatiotemporal transformation
- Representation as a weighted Difference of Gaussians (WDoG) kernel:

$$\varphi\ (x, t) = a\ (t)\ G_c(x) - \ b(t) G_s(x)$$

- $a(t)$ , $b(t)$ : time-varying weights which tune the shape of the DoG
- $\sigma_s$ , $\sigma_c$ : standard deviations of the center and the surround Gaussians respectively with $\sigma_c < \sigma_s$

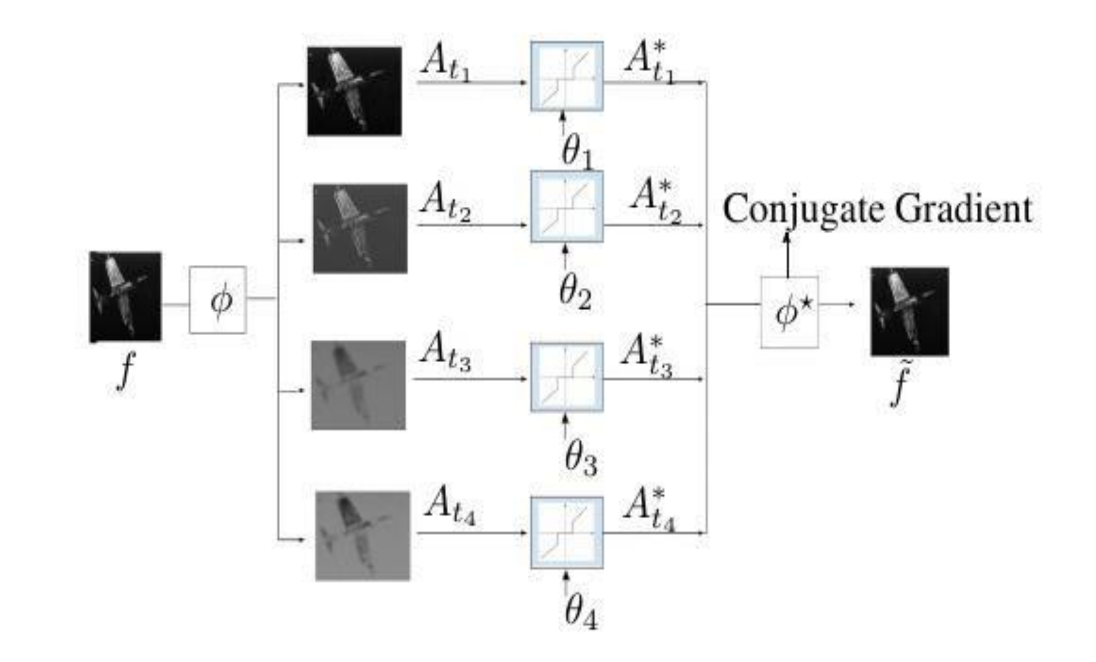


# Extending our System using the OPL

- The retina-inspired filtering, which is a frame, is applied to temporally constant input signals resulting in high redundancy:

$$A(x,t)=\varphi\,(x,t) * f\,(x)$$

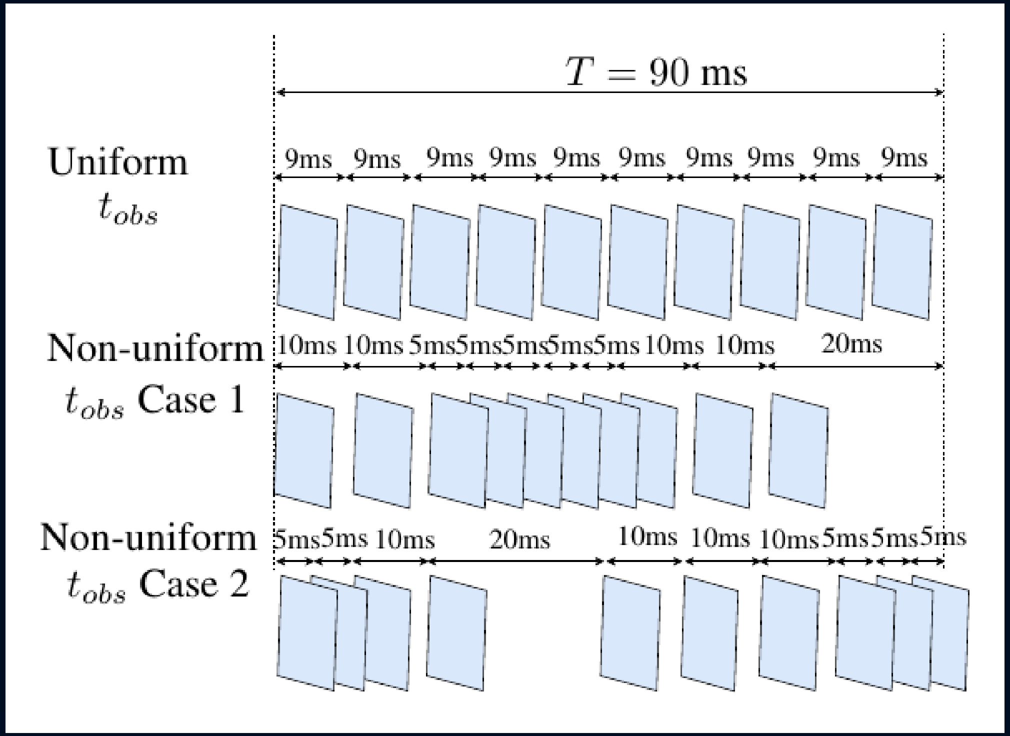
- In our extended system:
  1. OPL generates subbands removing spatial redundancies
  2. Then we quantize each subband generated by the OPL using our LIF Quantizer .




- The final output is the reconstruction using the subbands that have been encoded using the LIF.

# Finding the good subband generation

- Three different rates of subband generation
  - ➢ Uniform
  - ➢ Denser in the band-pass middle region of the OPL
  - ➢ Sparser in the band-pass middle region of the OPL
- We stop at observation time $t_{obs} = 90ms$. Later subbands are very redundant
- Subband generation for the non-Uniform cases is done experimentally without any specific function as it is only a first experimental approach



# Results on the OPL (1)

Original Image

Uniform generation

PSNR= 17.0814 dB
SSIM= 0.5204
Entropy= 3.316 bpp

Denser middle

PSNR= 15.1268 dB
SSIM= 0.4635
Entropy= 4.704 bpp

Sparser middle

PSNR= 24.7936 dB
SSIM= 0.8187
Entropy= 3.1 bpp

# Results on the OPL (2)

Original Image

Uniform generation

PSNR = 14.7250 dB
SSIM = 0.4843
Entropy = 4.769 bpp

Dense middle

PSNR = 19.7819 dB
SSIM = 0.7204
Entropy = 4.592 bpp

Sparse middle

PSNR = 20.4562 dB
SSIM = 0.7384
Entropy = 6.611 bpp

# Conclusions

- The LIF Quantizer is a very promising  innovative method for dynamic data encoding unlike the static existing methods.

- The refractory period introduces overload noise which yields the existence of an optimal threshold value θ that minimizes the MSE.

- Reduction of spatial redundancy can be achieved using the OPL filtering. This is  an extended more realistic model which better represents the retinal structure.
- Our first experimental attempt to find the good subband generation showed that the optimal choice depends on the image characteristics.
- A crucial next step on this study  is to develop a function for finding the best subband generation.

# References


- [1]  Gerstner, W. and Kistler, W. (2002). Spiking Neuron Models: An Introduction. Cambridge University Press, New York, NY, USA.
- [2] K. Masmoudi (2012). Retina-Inspired Image Coding Schemes. Université Nice Sophia Antipolis.
- [3] E. Doutsi, L. Fillatre, M. Antonini, and J. Gaulmin, “Retina-inspired filtering for dynamic image coding,” IEEE International Conference in Image Processing (ICIP), pp. 3505 – 3509, 2015
- [4] M. Dimopoulou, M. Antonini "Signal Quantization using a Leaky Integrate-and-Fire neuron", GRETSI 2017
- [5] H. Kolb (2004). How the Retina Works. American Scientist the magazine of Sigma Xi, The Scientific Research Society.
- [6] Nassi, J. J.,  Callaway, E. M. (2009). Parallel Processing Strategies of the Primate Visual System. Nature Reviews. Neuroscience, 10(5), 360-372


Thank you for
your attention!
Questions?