Divisive Normalization in V1 Neurons Through the Lens of Statistical and Information-Theoretic Approaches

Based on Schwartz & Simoncelli (1999) and Valerio & Navarro (2003)

Presented by:
O. Alalousi, M. Ben Kiran, C. Sarrau, I. Zughyer
Cognitive Sciences (CogSUP) Master 2, Computational Neuroscience and Artificial Intelligence track
November 2025

Summary

- Characteristics of natural images
- Non-linear processing of natural images
- Divisive Normalization Model: Statistical and Information Theoretic approach

  Simoncelli & Schwartz (1999)   Valerio & Navarro (2003)
- Results

Introduction:
Characteristics of natural images

Natural images are structured

- Highly **correlated** features
- **Information Redundancy**

=> Some features/pixels can be predicted by others

For example: Spatial redundancy (Nearby pixels share information)

=> Spatial correlation

Random pixel sampling : 100% 70% 60% 50% 40% 30% 20%

Relative loss of information

Luo et al. 2023

White noise has no structure

Random pixel sampling

50%

Total loss of information

Property used to compress natural images without losing information:

Some filtering

- ≈50%

50%

No information loss

Sensory Transduction aims to Reduce Redundancy

World $\xrightarrow{\text{Sensory Transduction}}$ Neural code

S (stimulus) $P_{R|S}$ R (Neural Response)

Redundancy Reduction Principle (Barlow 1961):
Early sensory processing goal = reduce redundancy of sensory encoding to ensure efficient coding

In early sensory system:
Redundancy Reduction ⇔ Maximization of Information

Original PNG - 12 MB

Compressed JPEG - 5 MB

Natural Images $\xrightarrow{\text{Sensory Transduction}}$ Neural code

(high redundancy)

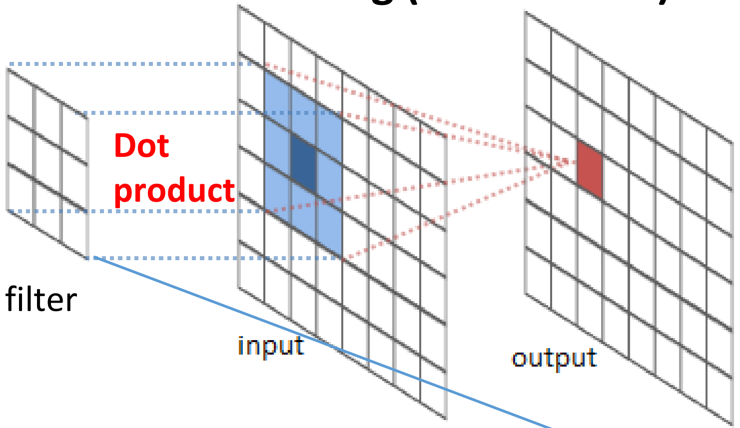
How can neural processing achieve this?

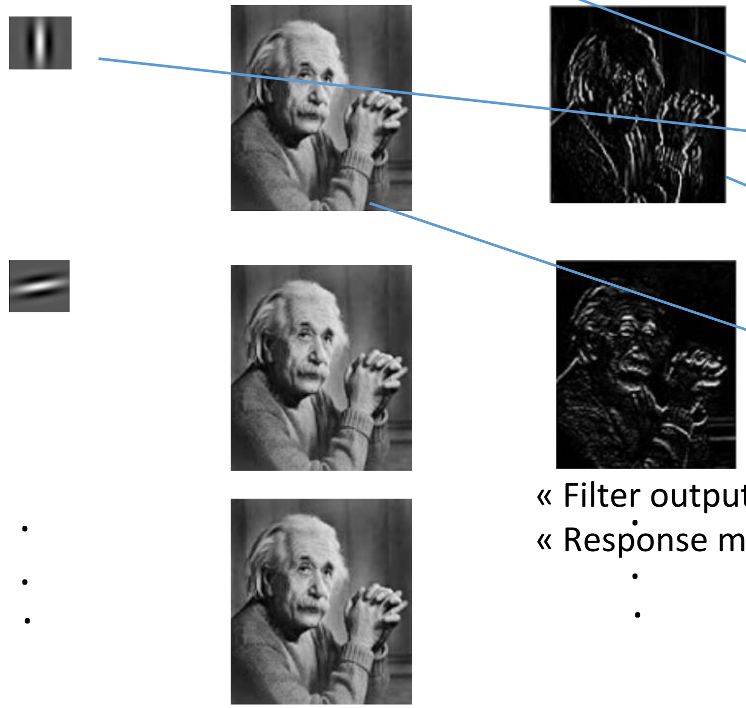
(No redundancy)

$$P(R_2|R_1) \approx P(R_2)$$
$$MI(R1; R2) \approx 0$$

Neurons do Linear Filtering

Linear filtering (convolution)





« Filter output »
« Response map »

Cosine Similarity

- If Filter matches scanned pixels
=> high positive output (white)
- No matching at all => output = 0

In neurons: Feedforward
Synaptic Weights

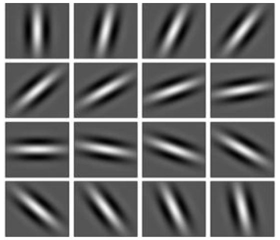
In neurons: **Firing rates**

In neurons: light intensity

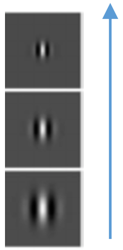
In neurons :
« Simple-cell
Receptive field »

Filters :

≠ Orientation

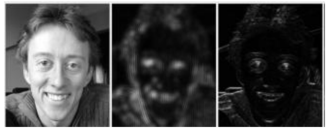


≠ Frequency (Scale)

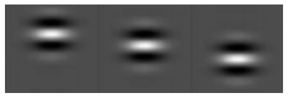


Finer details

Coarser details



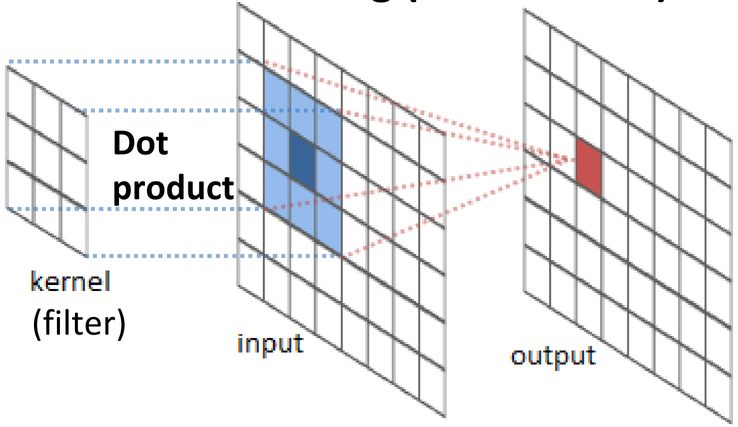
≠ Location



Each filter is *oriented, localized, and bandpass* at the same time

Neurons do Linear Filtering

Linear filtering (convolution)



Natural Images —Sensory Transduction→ Neural code

(high redundancy)

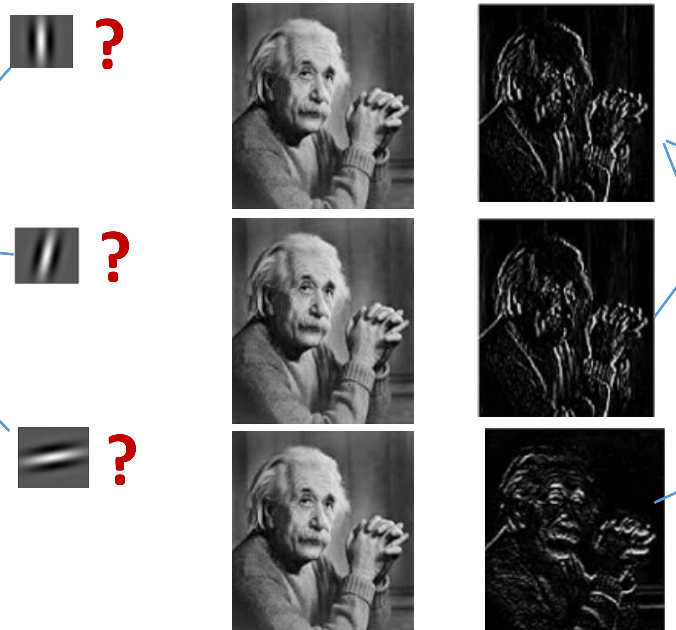
How can neural processing achieve this?

(No redundancy)

$$P(R_2|R_1) \approx P(R_2)$$
$$MI(R1;R2) \approx 0$$

Neurons try to find **filters (synaptic weights)** that decorrelate (= ↓ redundancy) natural images as much as possible

Synaptic Weights to be tuned to Natural Images Inputs



Redundancy => Tuning is not optimal

Less Redundancy

But .....

Is (optimal) neural linear filtering sufficient to reduce redundancy?

If neurons can learn to adapt their linear filters that decorrelate perfectly natural images (so reduce redundancy):

1- Would this optimal neural linear decorrelation (filtering) lead to the least redundancy (i.e. optimal coding) ?

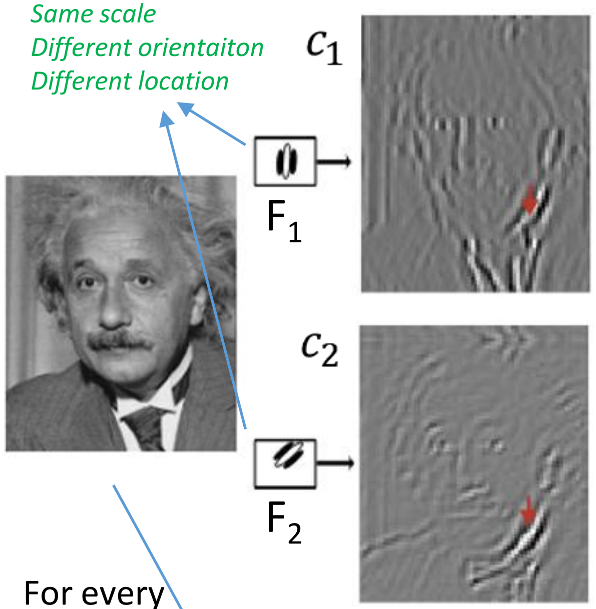
2- And why do sensory neurons show nonlinear behavior?

Non-linear processing of natural images

Natural images have complex nonlinear statistics!

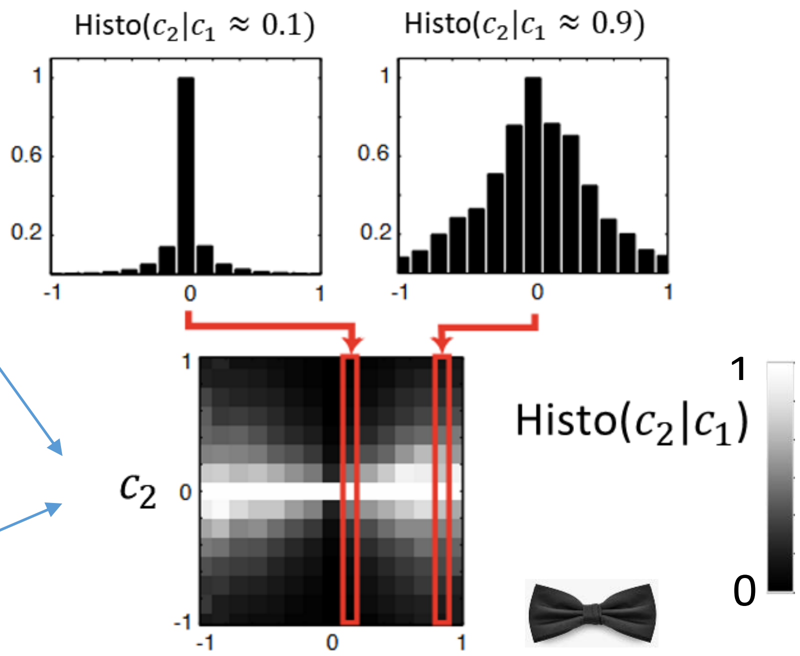
Linear filtering

Same scale
Different orientaiton
Different location



c = filter output

For every couple of pixels



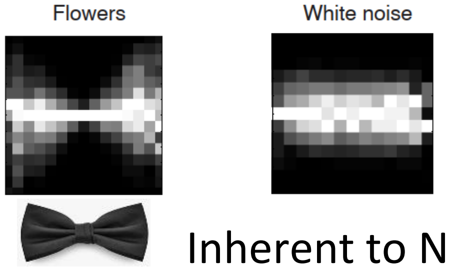
bowtie shape

Schwartz & Simoncelli (1999)

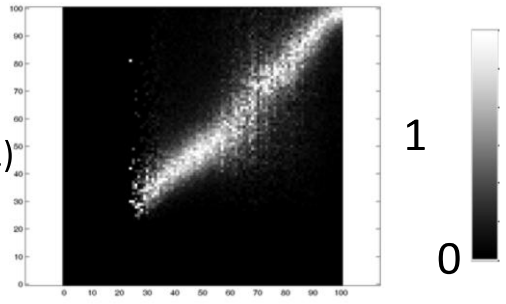
No correlation of the mean ≈ 0

But Var(c_2) ∝ c_1

⇒ There is a **correlation of the variance**



Inherent to Natural images



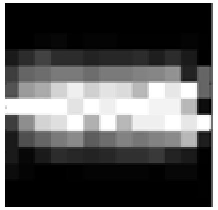
p2 (neighbor pixel to p1)

p1

histo($p_2|p_1$)

Strong correlation

If no correlation at all :



$$P(c_2|c_1) \approx P(c_2)$$
$$MI(c_2;c_1) \approx 0$$

Linear filtering decorrelated natural images only partially :
-> removed « mean correlation »
-> But still « variance correlation »
Renundancy persists

Neurons do linear filtering BUT insufficient to reduce redundancy

Natural Images $\xrightarrow{\text{Sensory Transduction}}$ Neural code

(high redundancy)

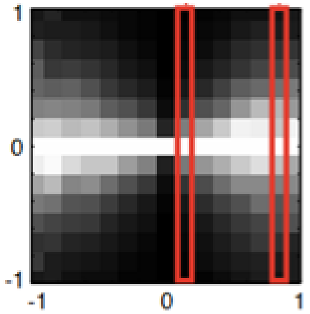
How can neural processing achieve this?

(No redundancy)

Neural Linear filtering is insufficient to have efficient coding

How to <u>statistically</u> perfectly decorrelate Natural Images?

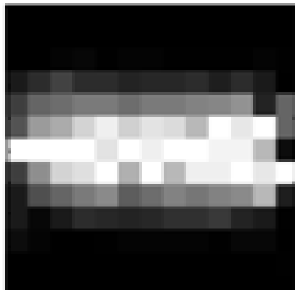
c_2



c_1

Histo$(c_2|c_1)$

?



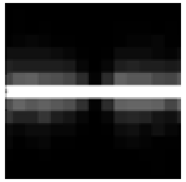
$$Var(c_2|c_1) \propto c_1$$

$$Var(c_2|c_1) \; = \; b_{12}\, {c_1}^2 \; + \; a^2$$

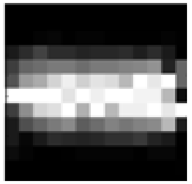
coefficient $\boldsymbol{b_{12}}$
Descibe how strong the dependency is between 2 neuron/filter outputs

High b

Lower b

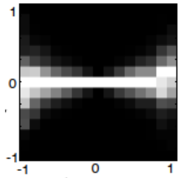


$b \; \approx 0$

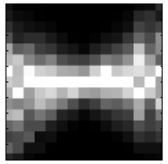


$b \; \approx 0$

Constant a^2
Var(c_2) if tnere is no correlation ($b = 0$)



a low



a high

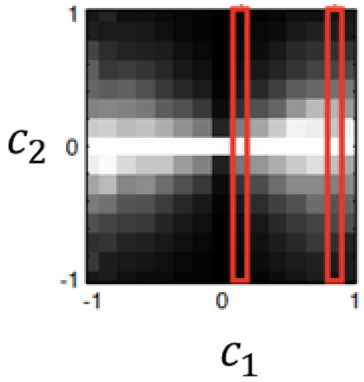
Dependancy on multiple neighbors

$$\mathrm{Var}\left(\mathrm{c}_i \middle| \{\mathrm{c}_j, j \in N_i\}\right) = \sum_j b_{ji} \mathrm{c}_j^2 + a^2$$

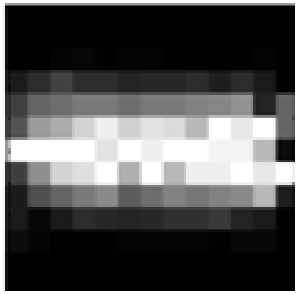
N_i = neighborhood around i^{th} neuron/filter

Relationship describing the remaining dependency (after linear filtering) on N_i neighbors neurons

How to statistically perfectly decorrelate Natural Images?



Histo($c_2|c_1$) **?**



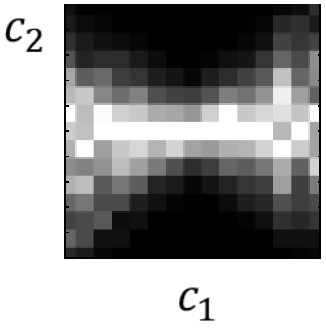
Divisive Normalization

<u>For dependency on 1 neighbor:</u>

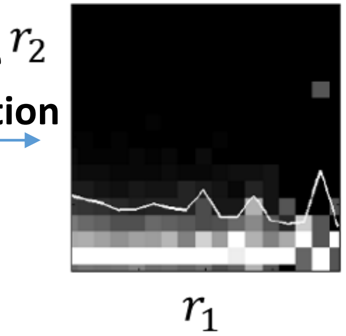
$$Var(c_2|c_1) = \ b_{12}\, {c_1}^2 + a^2$$

$$r_2 = \frac{c_2^2}{Var(c_2|c_1)}$$

$$r_2 = \frac{c_2^2}{b\, {c_1}^2 + a^2}$$



Divisive Normalization

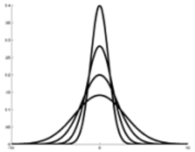


$$P(r_2|r_1) \approx P(r_2)$$
$$MI(r_2; r_1) \approx 0$$

<u>Normalized</u> outputs r1 & r2 became almost perfectly independant

<u>For dependency on N neighbors:</u>

$$r_i = \frac{c_i^2}{\sum_j b_{ji} c_j^2 + a^2}$$



Parameters fitted from natural images data (MLE by considering $Var(c_2, c_1)$ to be Gaussian)

1- Applying a **nonlinear** transformation (**divisive normalization**) -> would remove the remaining variance dependency
2- Parameters b & a are **learnt from Natural Images statistics**

Divisive Normalization Model: Statistical and Information Theoretic approach

Neurons do divisive normalization

Natural Images

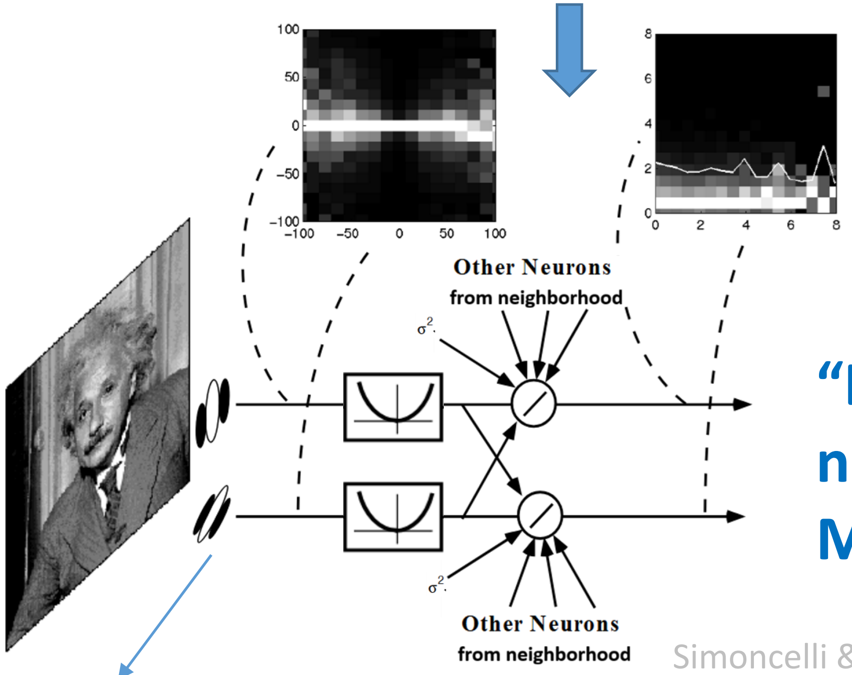
(high redundancy)

Sensory Transduction →

How can neural processing achieve this?

Neural code

(No redundancy)



"Divisive normalization" Model

Simoncelli & Schwartz 1999

1st step: Linear filtering
(Simple-cell Receptive Field)

=> Mean decorrelation

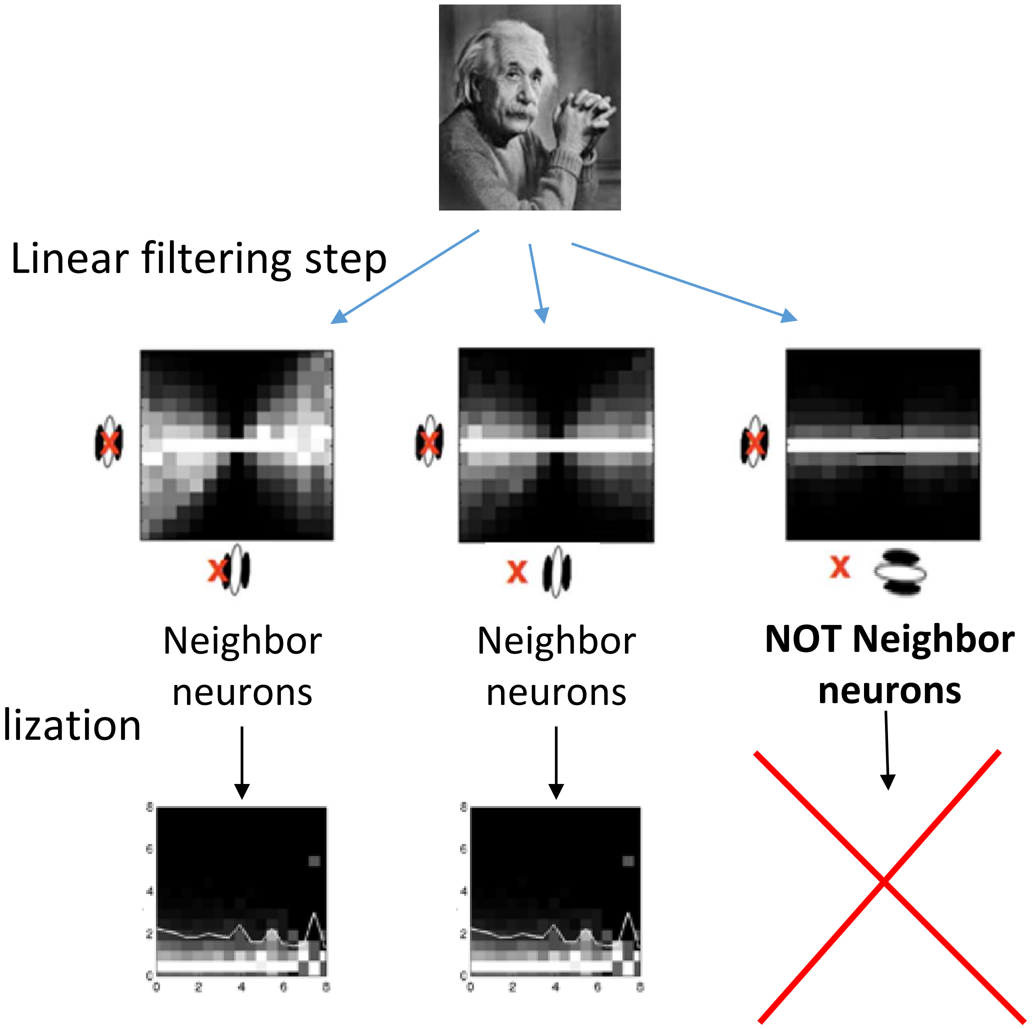
2st step: Divisive Normalization
(Gain control mechanism)

=> Variance decorrelation

"Divisive normalization" Model:
- Obtained by statistical optimization
- Found in real neurons (V1 cortex, auditory cortex, …)
- Leads to **efficient coding** (*Redundancy Reduction*)
- Explains interesting **observed physiology**
- **Canonical Computation** in the brain?

“Divisive Normalization” Model

Link to « horizontal feedback » and to « neighborhood » :
Firing rate of 1 neuron depends on firing rate of its neighboring neurons

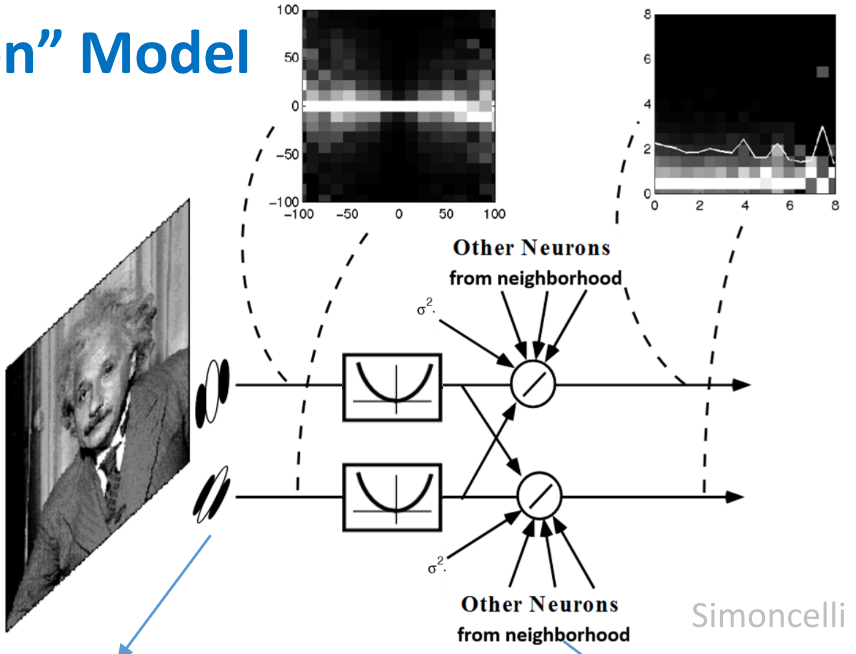


Divisive Normalization step

« Neighborhood » algorithmic meaning is also true at implementational microscopic level
-> Somatotopic organisation in the brain
-> Neurons that have close filters are close to each other
-> They send lateral connections to each other

=> No lateral feedback needed
(at least for the decorrelation issue)

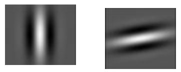
“Divisive Normalization” Model



Simoncelli & Schwartz 1999

1st step: Linear filtering
(Simple-cell Receptive Field)

Neural Network **learns** to tune its
feedforward synaptic weights (filter)
from visual stimuli

 …

-> Stable learning

2st step: Divisive Normalization
(Gain control mechanism)

Neural Network **learns** to tune its
lateral feedback weights
from visual stimuli

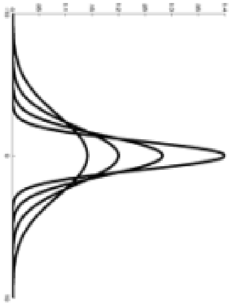
$$r_i = \frac{c_i^2}{\sum_j b_{ji} c_j^2 + a^2}$$

Dynamical learning
(changes with context)
=> Adaptive Learning

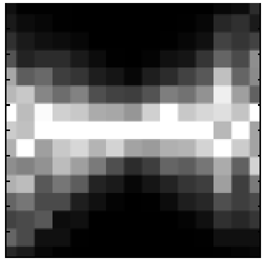
Theoretical-information driven parameters of the Divisive Normalization Model

Schwartz & Simoncelli (2001) (Statistical approach)

Proposed the Divisive Normalization Approach



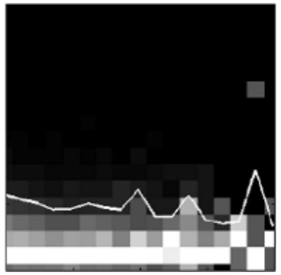
c_2



c_1

$$\frac{c_2^2}{\text{norm term}}$$

r_2



r_1

« Given 2 random variable, divising par condtional variance reduce the correlation »

$$r_2 = \frac{c_2^2}{Var(c_2|c_1)} = \frac{c_2^2}{bc_1^2 + a^2}$$

a and b describe the conditional variance

-> led to decorrelated outputs :

$P(r_2|r_1) \approx P(r_2)$

$MI(r_2;r_1) \approx 0$ -> this ***implicitly*** minimized MI

Fit of Conditional Variance parameters *(by matching data)*
-> Then consider them as Normalization parameters

Valerio & Navarro (2003) (Information-theoretic approach)

Adjustment to the Divisive Normalization Approach

Goal :
$\min MI(r_2;r_1)$
$\min MI(r_n;\ldots;r_2;r_1)$ (neurons 1 -> n being neighboring neurons)

Minimal $MI(r_2;r_1)$ is obtained when :

$$r_2 = \frac{c_2^2}{ec_1^2 + d^2}$$

=> They show that minimizing MI leads to the divisive normalization model !

But $e = b$ and $d = a$ does not give the lowest MI (only approximative solution)
So variance parameters (**Schwartz & Simoncelli**) are not the ones that guarantee best decorrelation

Fit of Normalization parameters (by minimizing MI)

Information-theoretic approach to Minimizing Redundancy

Conditional Variance $\mathrm{Var}(c_2|c_1) = bc_1^2 + a^2$

Normalization term $r_2 = \dfrac{c_2^2}{ec_1^2 + d^2}$

Minimization of MI

$$MI(r_2;r_1) = \int_0^{+\infty} p(r_1,r_2)\log\frac{p(r_1,r_2)}{p(r_1)\,p(r_2)}\,dr_1 dr_2$$

$$e, d^2 = \arg\min_{e,d^2} MI(r_2;r_1)$$

1) They assumed $\mathrm{Var}(c_2|c_1)$ to be Gaussian
$\Rightarrow$ Fitted a, b from data via MLE (as in Schwartz & Simoncelli)

2) Find e, d that $\min_{e,d^2} MI(r_2;r_1)$
(using many approximations: Taylor approximation, a & b as initial guess)

Generalization to multiple neighbors:

$$\mathrm{MI}(r_1, r_2, \ldots, r_n) = \int_0^{+\infty} \cdots \int_0^{+\infty} p(r_1, r_2, \ldots, r_n)\ \log\left(\frac{p(r_1, r_2, \ldots, r_n)}{p(r_1)\cdot p(r_2)\cdots p(r_n)}\right) \mathrm{d}r_1\,\mathrm{d}r_2\cdots\mathrm{d}r_n.$$

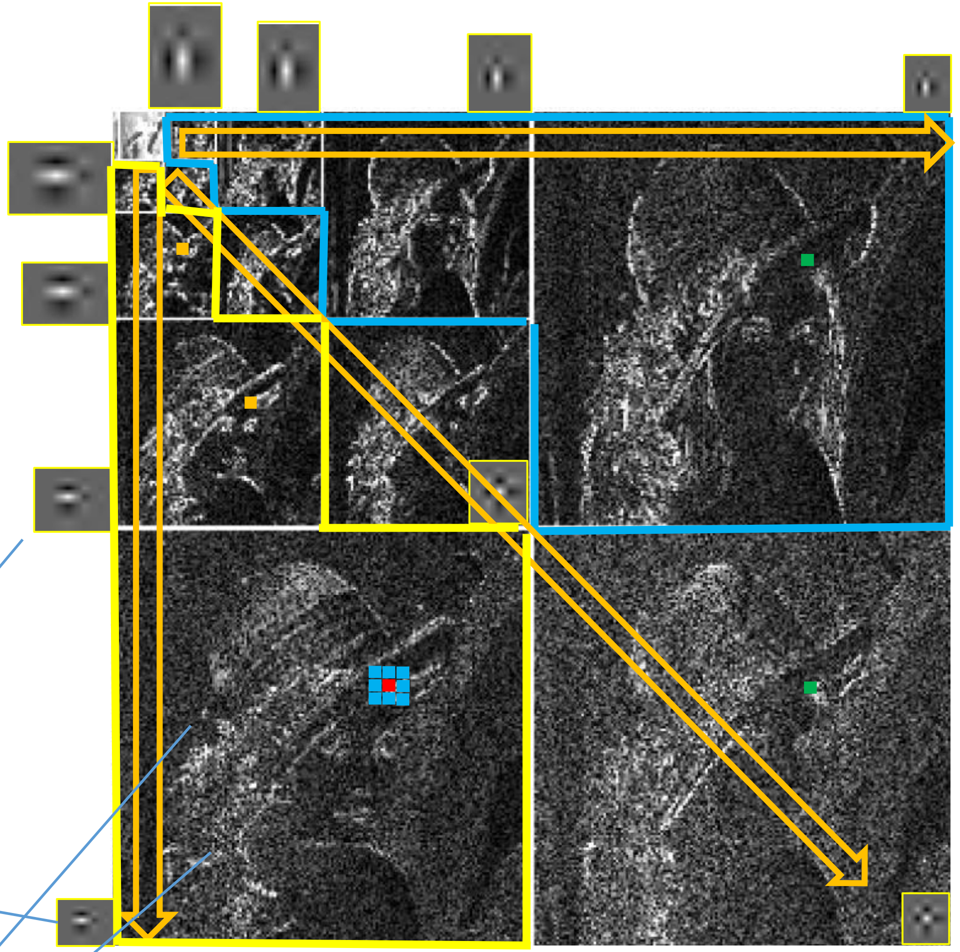
Results

6 black-and-white natural images

Linear filtering

(Daubechies filters)

In neurons: light intensity



Higher frequency filters (finer-detail detectors)

Vertical filters

Horizontal filters

Not surrounded = diagnoal filters

For each output c_i :

- 8 spatial neighbors
- 2 orientation neighbors
- 2 scale neighbors

In neurons: Synaptic weights (Receptive field when considering the spatial location too)

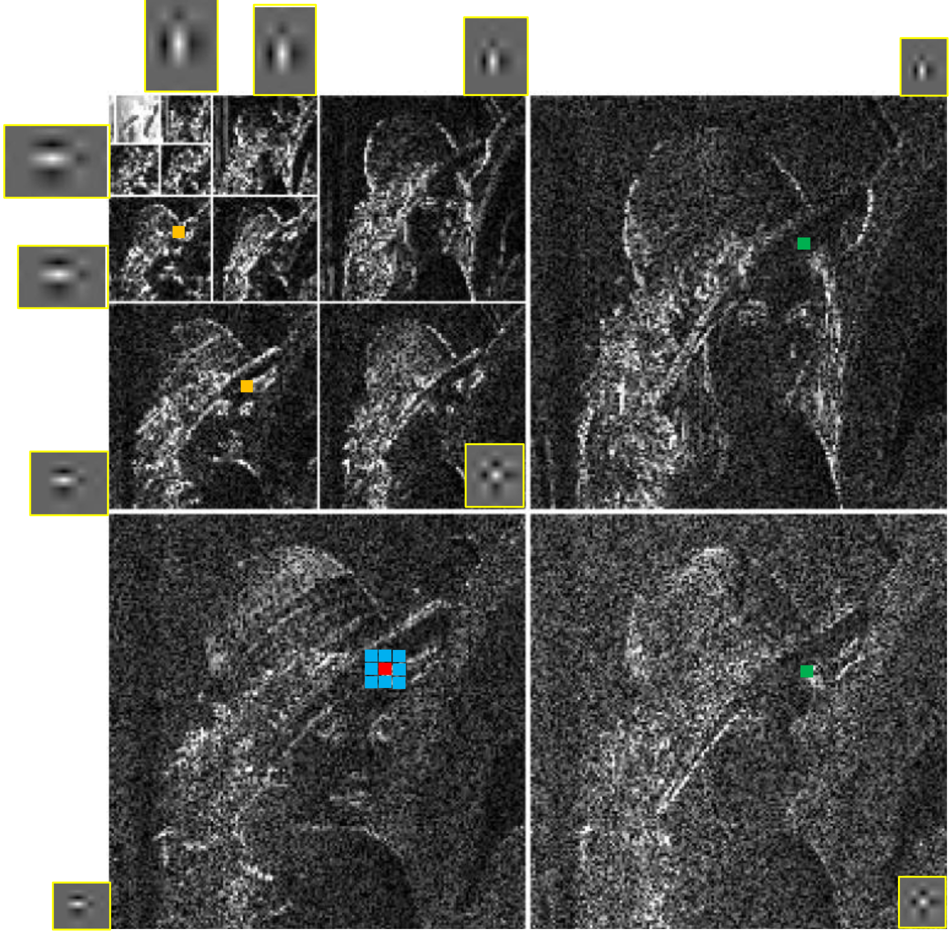
In neurons: Firing rates

Each case = 1 filter of some <u>orientation</u> and some <u>frequency</u> applied to ALL <u>locations</u>

Valerio & Navarro (2003)

6 black-and-white natural images

Linear filtering



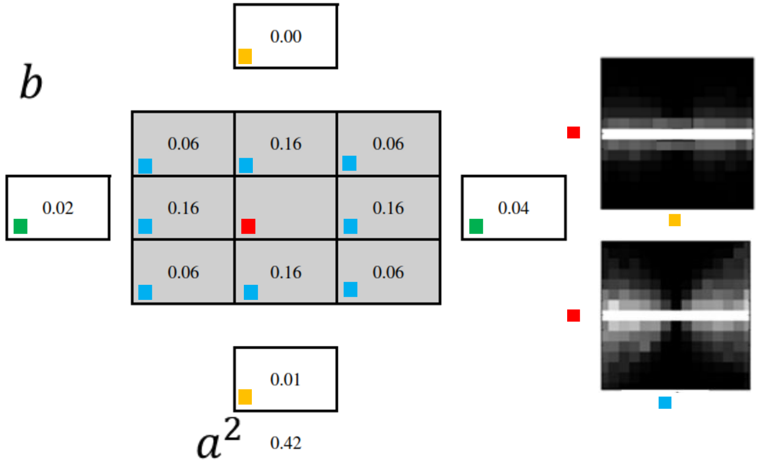
For each output $\boldsymbol{c_i}$:

- 8 spatial neighbors
- 2 orientation neighbors
- 2 scale neighbors

$$\mathrm{Var}(c_i|\{c_j\}) = \sum_{i=1}^{12} b_{ij} c_j^2 + a_i^2$$

$$P(c_i|\{c_j\}) = \frac{1}{\sqrt{2\pi\,\mathrm{Var}(c_i|\{c_j\})}} e^{-\frac{c_2^2}{2\,\mathrm{Var}(c_i|\{c_j\})}}$$

$$\{a_i\,, b_{ij}\} = \arg\max_{\{a_i\,, b_{ij}\}} \prod_{i,j} P(c_i|\{c_j\}) \qquad \textbf{(MLE)}$$



Parameters used by Schwartz & Simoncelli for normalization

$$r_i = \frac{c_i^2}{\sum_{i=1}^{12}(b_{ij} c_j^2) + a_i^2}$$

But minimizing Mutual information min $MI(r_n; \ldots; r_2; r_1)$ lead to other parameters :

$$\mathrm{Var}(c_i | \{c_j\}) = \sum_{i=1}^{12} (b_{ij} c_j^2) + a_i^2 \longrightarrow$$ Fitted by MLE from data

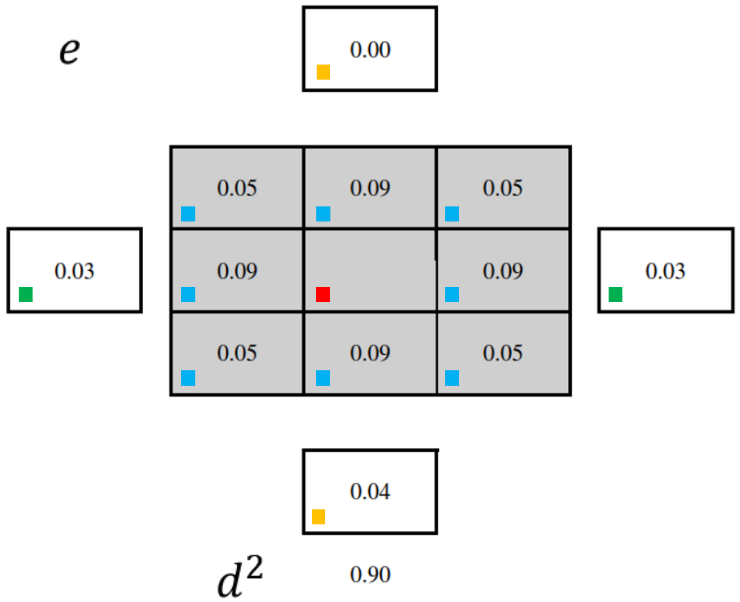
$$r_i = \frac{c_i^2}{\sum_{i=1}^{12} (e_{ij} c_j^2) + d_i^2}$$

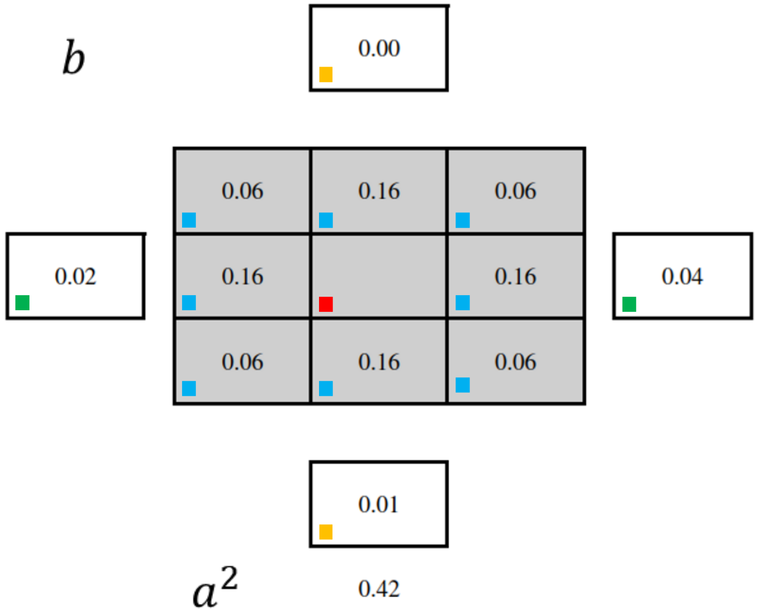
$$e, d = \underset{e,d}{\arg\min}\, MI(r_i ; \{r_j\}) \longrightarrow$$

Optimization by grid search

- Using many approximations : Taylor, fitted parameters a, b, ..

Valerio & Navarro (quasi-)optimal parameters



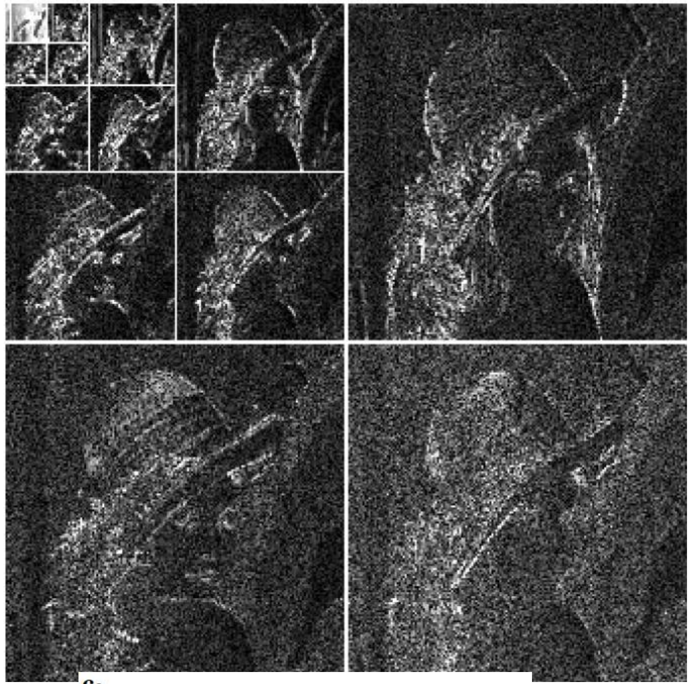


Schwartz & Simoncelli Approximated parameters

Putting it all together

1st step

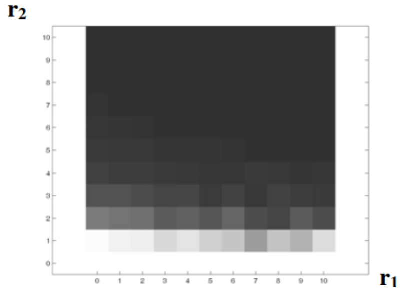
Linear filtering



2nd step

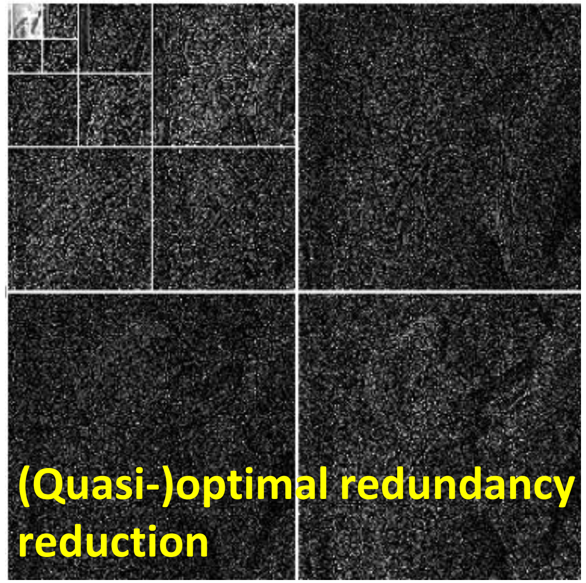
Divisive normalization
(over 12 neighbors)
using a, b

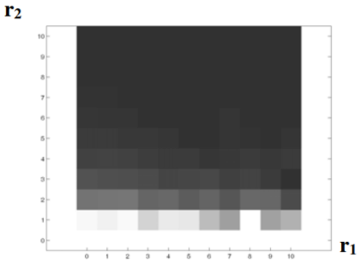
Response maps not shown



$MI\ (r_1; r_2) \approx 0.0113$
strong independance

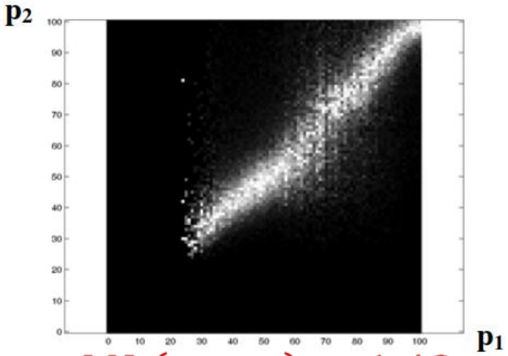
Divisive normalization
(over 12 neighbors)
using e, d (quasi-optimal parameters)



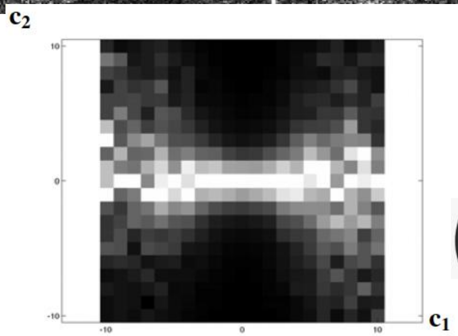


$MI\ (r_1; r_2) \approx 0.0101$
(−11%)
strong independance

p2 is a neighbor pixel to p1

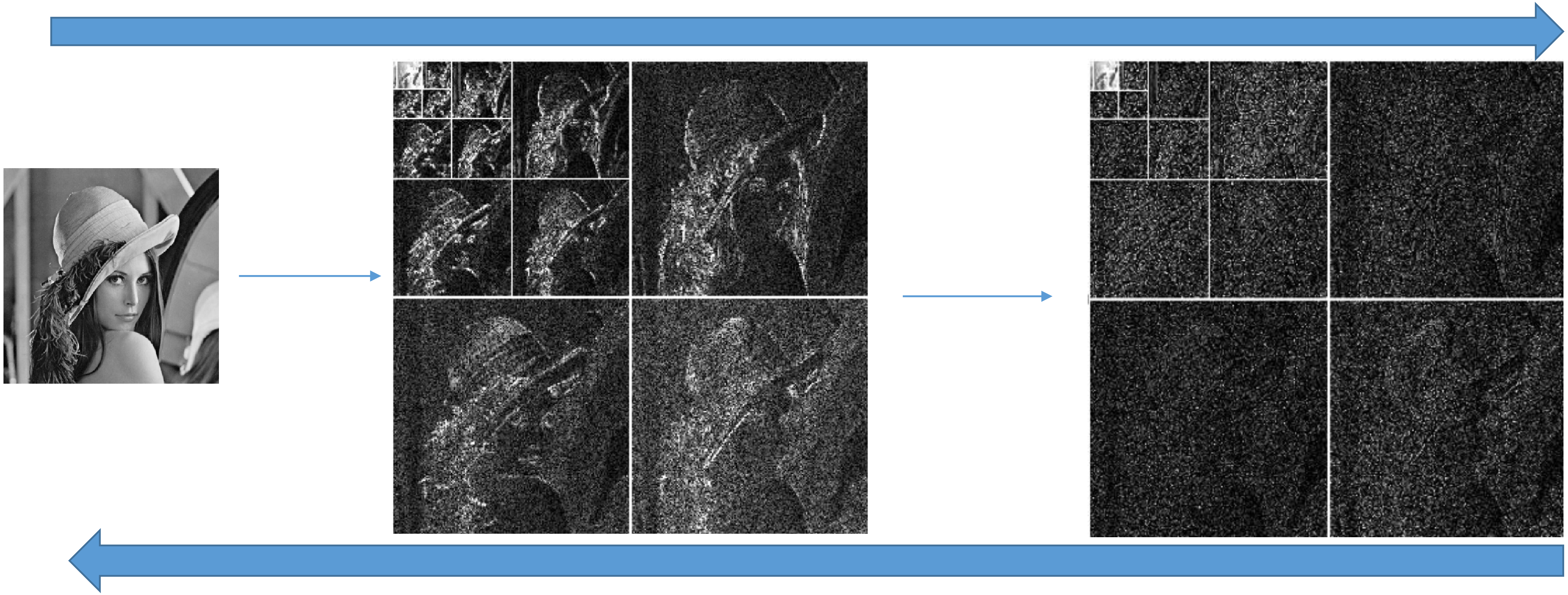


$MI\ (p_1; p_2) \approx 1.43$
Strong correlation



$MI\ (c_1; c_2) \approx 0.12$
Lower (but persistent) correlation

Invertibility of the Divisive Normalization Model



$$\begin{pmatrix} c_1^2 \\ c_2^2 \\ \cdots \\ c_n^2 \end{pmatrix} = (\mathbf{Id} - \mathbf{R} \cdot \mathbf{E})^{-1} \begin{pmatrix} r_1 d_1^2 \\ r_2 d_2^2 \\ \cdots \\ r_n d_n^2 \end{pmatrix}.$$

Store :

- **Linear filtering parameters**
- **Parameters *d*, *e***
- Sign of *c*

In neurons:

- **- Feedforward** synaptic weights
- **- Lateral Feedback** synaptic weights

+ (**Top-Down Feedback** from higher-level cortex) (Bayesian predictive coding)

Conclusion

- The Divisive Normalization Model **links the particular characteristics of natural images to efficient coding**
- The **information-theory** approach leads to a **quasi-optimal solution**, not achieved by the statistical one
- Decorrelates variance thanks to **lateral feedback** from neighboring neurons
- **Adaptive learning** (context-dependant)

Implications:

- It's **found in sensory systems in the brain** (V1 neurons, light adaptation and contrast normalization in the retina, auditory neurons).
- **Found in other species,** for example olfactory processing in the fruitfly antennal lobe
- Has been **adapted into some Artificial Neural Networks to improve image processing** and invariance to non-informative changes in images (ex. variations in contrast and illumination).

Discussion

Many questions remain:

- Do **brain computational areas other** than sensory transduction **(memory, language processing ..)** use a normalization processing?
- The model takes only into account the suppressing activity from neighboring neurons. **What about facilitatory lateral feedback?**
- They did found parameters from information-theoretic perspective. What about **tuned parameters** from **real recordings from V1 neurons** exposed to natural images ?
- Is finding **« <u>the</u> »** lowest mutual information **really what the early sensory biological networks try to do**?
- Is the brain capable of “perfectly” decorrelating the input? **Does it need to do it?** What about **prior from predictive coding**?
- At what pace does the neural visual system **adapt its lateral feedback weights**?