

Motion Invariance in Visual Environments

Alessandro Betti^{1,2}, Marco Gori² and Stefano Melacci²

¹University of Florence, Florence, Italy

²DIISM, University of Siena, Siena, Italy

alessandro.betti@unifi.it, {marco, mela}@diism.unisi.it

Abstract

The puzzle of computer vision might find new sophisticated solutions when we realize that most successful methods are working at image level, which is remarkably more difficult than processing directly visual streams, just as it happens in nature. In this paper, we claim that the processing of a stream of frames naturally leads to formulate the motion invariance principle, which enables the construction of a new theory of visual learning based on convolutional features. The theory addresses a number of intriguing questions that arise in natural vision, and offers a well-posed computational scheme for the discovery of convolutional filters over the retina. They are driven by the Euler-Lagrange differential equations derived from the principle of least cognitive action, that parallels the laws of mechanics. Unlike traditional convolutional networks, which need massive supervision, the proposed theory offers a truly new scenario in which feature learning takes place by unsupervised processing of video signals. An experimental report of the theory is presented where we show that features extracted under motion invariance yield an improvement that can be assessed by measuring information-based indexes.

1 Introduction

While the emphasis on a general theory of vision was already the main objective at the dawn of the discipline [Marr, 1982], computer vision has evolved without a systematic exploration of foundations in the framework of machine learning. In particular, in most cases, computer vision is regarded just as an application of machine learning. When the target is moved to unrestricted visual environments and the emphasis is shifted from huge labelled databases to a human-like protocol of interaction, we need to go beyond the current peaceful interlude that we are experimenting in vision and machine learning. So far, the semantic labeling of pixels of a given video stream has been mostly carried out at frame level. This seems to be the natural outcome of well-established pattern recognition methods working on images, which have given rise to nowadays emphasis on collecting big labelled image

databases (e.g., [Deng *et al.*, 2009]) with the purpose of devising and testing sophisticated machine learning algorithms.

A crucial problem that has been recognized by Poggio and Anselmi [Poggio and Anselmi, 2016] is the need to incorporate visual invariances into deep nets that go beyond simple translation invariance that is currently characterizing convolutional networks. They propose an elegant mathematical framework on visual invariance and enlighten some intriguing neurobiological connections. Overall, the ambition of extracting distinctive features from vision poses a challenging task. While we are typically concerned with feature extraction methods that are independent of classic geometric transformation, it looks like we are still missing the fantastic human skill of capturing distinctive features to recognize “ironed and rumpled shirts”, for example. There is no apparent difficulty to recognize shirts by keeping the recognition coherence in case we roll up the sleeves, or we simply curl them up into a ball for the laundry basket. Of course, there are neither rigid transformations, like translations and rotation, nor scale maps, that transforms an ironed shirt into the same shirt thrown into the laundry basket. In this paper, we claim that motion invariance can in fact capture all we need. Translation and scale invariance, that have been the subject of many studies [Lowe, 2004; Gori *et al.*, 2016], are in fact examples of invariances that can be fully gained whenever we develop the ability to detect features that are invariant under motion. For instance, the moving of the finger experimented by infants leads them to enforce a natural invariance: The finger will become bigger and bigger as it approaches their face, but it is still their inch, which requires to impose a consistent decision. Clearly, translation, rotation, and complex deformation invariances derive from motion invariance. Humans always experiment motion in their life, so as the gained visual invariances naturally arise from motion invariance. Animals with foveal eyes also move quickly the focus of attention when looking at fixed objects, which means that they continually experiment motion. Hence, also in case of fixed images, conjugate, vergence, saccadic, smooth pursuit, and vestibulo-ocular movements lead to acquire visual information from relative motion. We claim that the production of such a continuous visual stream naturally drives feature extraction, since the corresponding convolutional features are expected not to change during motion. The enforcement of this consistency condition

creates a mine of visual data during animal life. Of course, we need to compute the optical flow at pixel level so as to enforce the consistency of all the extracted features. Early studies on this problem [Horn and Schunck, 1981], along with recent related improvements (see e.g. [Baker *et al.*, 2011]) suggest to start computing the velocity field by enforcing brightness invariance. As the optical flow is gained, it is used to enforce motion consistency on the visual features. Interestingly, the theory we propose is quite related to the variational approach that is used to determine the optical flow in [Horn and Schunck, 1981], so that the developed features could be also used to reinforce motion estimation. It is worth mentioning that an effective visual system must also develop features that do not follow motion invariance. What we study in this paper is general enough to embrace also the case in which some of the learned features are not subject to motion invariance. These kind of features can be conveniently combined with those that are discussed in this paper with the purpose of carrying out high level visual tasks.

The visual features that we propose in this paper are derived in the framework of the principle of cognitive action [Betti and Gori, 2016], which gives rise to a time-variant differential equation, where the Lagrangian coordinates correspond with the values of the convolutional filters. The learning process can be interpreted in the framework of the minimization of the cognitive action that offers a self-consistent framework. There are other works that study the unsupervised learning of visual features for temporally varying signals, such as Slow Feature Analysis (SFA) [Wiskott and Sejnowski, 2002; Sun *et al.*, 2014]. SFA includes motion estimation as an example of application of SFA itself, while here we focus on exploiting motion to learn pixel-level features. In the last decade, a number of approaches have been proposed to develop features in an unsupervised manner, mostly using collections of images [Huang *et al.*, 2007; Kavukcuoglu *et al.*, 2010]. Only in the last few years we find some works that consider unsupervised learning from video data, jointly with motion [Wang and Gupta, 2015; Li *et al.*, 2016; Pathak *et al.*, 2017]. These works usually learn image-level representations, and, to the best of our knowledge, none of them aims at developing a learning theory for the unsupervised development of visual features that is specifically rooted around the notion of motion, that is the goal of this paper.

This paper is organized as follows. Section 2 introduces the principle of least cognitive action and the basic elements of the proposed architecture. Section 3 focusses on the discrete case, and defines the differential equations that we integrate to develop features while processing the video stream. Section 4 includes a collection of experiments aimed at showing the behaviour of the system under different configurations, in terms of information-based indices. Finally, Section 5 concludes the paper. Supplementary Material can be found at <http://sailab.diism.unisi.it/motion-invariance/>.

2 The Principle of Least Cognitive Action

We consider the mechanisms that give rise to the construction of local features for any pixel $x \in X$ of the retina, at any

time $t \in [0, T]$. These features, along with the video itself, therefore can be regarded as visual fields, that are defined on the domain $D = X \times [0, T]$. A set of symbols are extracted at every layer of a deep architecture, so as each pixel—along with its context—turns out to be represented by the list of symbols extracted at each layer.

In what follows, points on the retina will be represented with two dimensional vectors $x = (x_1, x_2)$. The temporal coordinate is usually denoted by t , and, therefore, the video signal on the pair (x, t) is $C(x, t)$. This *color field* can be thought of as a field that is characterized by m components for each single pixel ($m = 3$ for RGB). We are concerned with the problem of extracting visual features that, unlike the components of the video, express the information associated with the pair (x, t) and with the neighborhood of pixel x . A possible way of constructing this kind of features is to define

$$\mathcal{C}_i(x, t) = \frac{1}{n} + \sum_{j=1}^m \int_X dy \varphi_{ij}(x, y, t) C_j(y, t) \quad (1)$$

Here we assume that n symbols (indexed by i) are generated from the m components of the video. Notice that the kernel $\varphi(x, y, t)$ is responsible of expressing the spatial dependencies. It is worth mentioning that whenever $\varphi(x, y, t) \rightsquigarrow \varphi(x - y, t)$ the above definition reduces to an ordinary spatial convolution. The computation of $\mathcal{C}_i(x, t)$ yields a field with n features, and Eq. (1) can be used for carrying out a piping scheme where a new set of features \mathcal{C}' is computed from \mathcal{C} . Of course, this process can be continued according to a deep computational structure with a homogeneous convolutional-based computation, which yields the features \mathcal{C} at the z -th convolutional layer. The theory proposed in this paper focuses on the construction of any of these convolutional layers which are expected to provide higher and higher abstraction as we increase the number of layers. The *filters* φ are what completely determines the features \mathcal{C} . In this paper we formulate a theory for the discovery of φ that is based on three driving principles, that are described below.

Optimization of information-based indices. Beginning from the color field C , we attach a symbol $y_i \in \Sigma$ of a discrete vocabulary to pixel (x, t) with probability $\mathcal{C}_i(x, t)$, assuming that $\forall x, t : \mathcal{C}_i(x, t)$ is subject to the probabilistic constraints $\sum_i \mathcal{C}_i(x, t) = 1$ (normalization) and $0 \leq \mathcal{C}_i(x, t) \leq 1$ (positivity). The quantity $\mathcal{C}_i(x, t)$ can be identified with the conditional probability of the random variable Y associated with the symbols y_i subject to a certain value of the random variables X, T and F that models, respectively, the distribution of the position over the retina, of the temporal coordinate and of all the possible configurations of pixel intensities over the frame. The conditional entropy $S(Y | X, T, F)$ is given by $S(Y | X, T, F) = - \int_{\Omega} \sum_{i=1}^n dP_{X,T,F} p_i \log p_i$ where p_i is the conditional probability of Y conditioned to the values of X, T and F , $dP_{X,T,F}$ is the joint measure of the variable X, T, F , and Ω is a Borel set in the (X, T, F) space. We can rewrite the conditional entropy as $S(Y | X, T, F) = - \int_D d\mu(x, t) \sum_{i=1}^n \mathcal{C}_i(x, t) \log \mathcal{C}_i(x, t)$, where $\mu(x, t)$ is a space-time measure. Clearly, we want to keep the conditional entropy as small as possible so as to develop dominating features. At the same time we must ensure that the entropy

of variable Y , $S(Y) = -\sum_{i=1}^n \Pr(Y = y_i) \log \Pr(Y = y_i)$, must be as high as possible, since this ensures the development of all the features associated with the alphabet of symbols. If we use the law of total probability to express $\Pr(Y = y_i)$ in terms of the conditional probability p_i and use the above assumptions we get $\Pr(Y = y_i) = \int_{\Omega} dP_{X,T,F} p_i = \int_D d\mu(x,t) \mathcal{C}_i(x,t)$. Then $S(Y) = -\sum_{i=1}^n (\int_D d\mu(x,t) \mathcal{C}_i(x,t)) \cdot \log(\int_D d\mu(x,t) \mathcal{C}_i(x,t))$. To sum up (see the Supplementary Material for further details on the computation of $S(Y)$), the index

$$\mathcal{I}(\varphi) = S(Y) - S(Y | X, T, F),$$

which is somewhat related to the classic Shannon mutual information, must be maximized [Gori *et al.*, 2012; Melacci and Gori, 2012].

Motion invariance. If we focus attention on a pixel x at time t , which moves according to the trajectory $x(t)$ then $\mathcal{C}(x(t), t) = c$, being c a constant. This ‘‘adiabatic’’ condition is thus expressed by the condition $d\mathcal{C}/dt = 0$, which yields

$$\mathcal{M}(\varphi) := \partial_t \mathcal{C}_i + \sum_{j=1}^2 v_j \partial_j \mathcal{C}_i = 0, \quad (2)$$

where $v: D \rightarrow \mathbf{R}^2$ is the velocity field that we assume to be given, and ∂_k is the partial derivative with respect to x_k . When replacing \mathcal{C}_i as stated by Eq. (1) we get

$$\mathcal{M}(\varphi) = \sum_{j=1}^m \int_X dy (\partial_t \varphi_{ij} C_j + \varphi_{ij} \partial_t C_j + \sum_{k=1}^2 v_k \partial_k \varphi_{ij} C_j),$$

which holds for any $(t, x) \in D$. Notice that this constraint is linear in the field φ . This can be interpreted by stating that learning under motion invariance consists of determining elements of the kernel of the function $\mathcal{M}(\varphi)$. Clearly, the learning process is expected to keep the value of $\mathcal{M}(\varphi)$ as small as possible.

Parsimony principle. Like any principled formulation of learning, we require the filters to obey the parsimony principle. Amongst the philosophical implications, it also favors the development of a unique solution. Given the filters φ , there are two parsimony terms, one $\mathcal{P}(\varphi)$, that penalizes abrupt spatial changes, and another one, $\mathcal{K}(\varphi)$ that penalizes quick temporal transitions. Ordinary regularization issues suggest to discover functions φ_{ij} such that

$$\lambda_P \mathcal{P} + \lambda_K \mathcal{K} = \sum_{\substack{1 \leq i \leq n \\ 1 \leq j \leq m}} \int_D dt dx h(t) \left[\frac{\lambda_P}{2} (P_x \varphi_{ij}(x, t))^2 + \frac{\lambda_K}{2} (P_t \varphi_{ij}(x, t))^2 \right],$$

is ‘‘small’’, where P_x, P_t are spatial and temporal differential operators, and λ_P, λ_K are non-negative reals. We assumed an ergodic translation of $d\mu$, that, in this case, only involves the temporal factor $h(t)$.

Overall, the process of learning is regarded as the minimization of the *cognitive action*

$$\mathcal{A}(\varphi) = -\mathcal{I}(\varphi) + \lambda_M \mathcal{M}(\varphi) + \lambda_P \mathcal{P}(\varphi) + \lambda_K \mathcal{K}(\varphi), \quad (3)$$

where $\lambda_M, \lambda_P, \lambda_K$ are positive multipliers. While the first and third principles are typically adopted in classic unsupervised learning, motion invariance does characterize the approach followed in this paper. The motion invariance principle can also be limited to a subset of the n features. In other words, our model can also develop visual features that do not obey the motion invariance principle. Basically, the process of learning consists of solving the variational problem $\hat{\varphi} = \arg \min_{\varphi} \mathcal{A}(\varphi)$ (see the Supplementary Material for further details). As it will be shown in the following, in our multi-layer implementation the minimization of $\mathcal{A}(\cdot)$ takes place at each layer of the architecture, involving the filters of the considered layer only, relying on a piping scheme that is inspired to developmental learning issues.

3 Euler-Lagrange Equations

The field theory of the previous section can be approximated over the discrete (and bounded) retina X^\sharp , where the video frames are represented. Then, the Euler-Lagrange equations of the cognitive action (3) lead to those differential equations that we can integrate to learn the convolutional filters.

In detail, instead of the fields $\varphi_{ij}(x, t)$, we consider a bunch of functions of time $\varphi_{ijx}(t)$, indexed by the point on the retina x other than the filter/feature index i and the input channel index j . Similarly, the color field will be replaced by $C_{kx}(t)$. In doing so the motion invariance becomes a quadratic form in φ and $\dot{\varphi}$, while the other relevant terms of the theory (entropy, relative entropy) are trivially functions of $\varphi(t)$ and t .

We assume filters to have a unique and finite size for all the features. As a consequence, for each feature i , we can flatten the filter φ_{ijx} into a vector, and concatenate the n filter-vectors into q . We selected a second order term to implement the parsimony principle, $h/2(\alpha|\dot{q}|^2 + \beta|q|^2 + 2\gamma\dot{q} \cdot \dot{q} + k|q|^2)$, being α, β, γ and k positive constants. If we make the entropy term local in time, and evaluate the first variation of the discretized cognitive action, the differential Euler-Lagrange (EL) equations are (for the sake of simplicity, we skip the derivations, see Supplementary Material for all the details):

$$\hat{\alpha}q^{(4)} + 2\hat{\alpha}q^{(3)} + (\hat{\alpha} + \hat{\gamma} - \hat{R})\ddot{q} - \left(\hat{R} - \hat{\gamma} - \lambda_M \hat{N}^\sharp + \lambda_M (\hat{N}^\sharp)' \right) \dot{q} - \left((\hat{N}^\sharp)' - \hat{Z} \right) q + \nabla_q w = \frac{\lambda_C - 1}{n} b, \quad (4)$$

where $q^{(4)}, q^{(3)}$ are the fourth and third derivatives of q over time, λ_C is a positive constant, and we have used the notation $\hat{f} = hf$ (so that for example $\hat{\alpha} = \hbar\alpha$). In order to define the other terms, we introduce the notation Γ_x to indicate the area (volume if $m > 1$) of the input signal centred around x , of the same size of the filters, flattened into a vector. We have $R := \beta + \lambda_M M^\sharp$, and the notation A^\sharp indicates a block-diagonal matrix whose blocks are A . The matrix M is composed of $M_{al} := \sum_{x \in X^\sharp} g_x \Gamma_x(a) \Gamma_x(l)$, and g_x is a distribution over the retina. Analogously, we define $N_{al} := \sum_{x \in X^\sharp} g_x r(a) \Gamma_x(l)$, being $r := \hat{\Gamma}_x + v \cdot \Delta \Gamma_x$. We have $Z := k + B^\sharp - \lambda_C M^\sharp + \lambda_1 \tilde{M} + \lambda_M O^\sharp$, where \tilde{M} is a squared matrix composed of $m \times m$ repetitions of M , λ_1 is a positive

constant, $B_{al} = b_a b_l$, and $b_a = \sum_{x \in X^\#} g_x \Gamma_x(a)$. The matrix O is composed of $O_{al} = \sum_{x \in X^\#} g_x r(a) r(l)$. Finally, $w(t, q) := \sum_{x \in X^\#} g_x (\frac{1}{n} + \xi(q, \Gamma_x))' [\frac{1}{n} + \xi(q, \Gamma_x) < 0]$, where $\xi(q, \Gamma_x)$ returns the n -length vector with the result of the convolutions of the n filters with the input, $[\cdot] = 1$ if the condition in brackets is true, otherwise it is 0, and it operates element-wise when a vector of conditions is provided.

In deriving equations some conditions arises naturally at $t = T$ (see the Supplementary Material for more details):

$$\begin{aligned} \hat{\alpha} \ddot{q} + \hat{\gamma} \dot{q} &= 0, \\ \hat{\alpha} q^{(3)} + \hat{\alpha} \dot{q} - (\hat{\beta} + \lambda_M \hat{M}^\# - \hat{\gamma}) \dot{q} - \lambda_M (\hat{N}^\#)' q &= 0. \end{aligned} \quad (5)$$

An interesting special case of these equations is that obtained with a null signal $C \equiv 0$. With this assumption and assuming that $h(t) = e^{\theta t}$, $\theta > 0$, our equations (4) become

$$\alpha q^{(4)} + 2\theta \alpha q^{(3)} + (\theta^2 \alpha + \theta \gamma - \beta) \ddot{q} + (\theta^2 \gamma - \theta \beta) \dot{q} + kq = 0. \quad (6)$$

In order to see whether this equation can be stable we need to apply the Routh-Hurwitz criterion. For a fourth order ODE $q^{(4)} + aq^{(3)} + bq'' + cq' + dq = 0$ this criterion reduces to check if $a > 0$, $b > 0$, $0 < c < ab$ and $0 < d < (abc - c^2)/a^2$ that in our case means that

$$\gamma > \frac{\beta}{\theta}, \quad k < \frac{(\beta - \gamma\theta)[\beta - \theta(\gamma + 2\alpha\theta)]}{4\alpha}. \quad (7)$$

So for example if we choose $\alpha = k = 1/2$, $\gamma = 2$ and $\theta = \beta = 1$ we obtain a stable equation. This being said it is also crucial to notice that we have control over the important parameter of the theory θ as long as we choose the regularization parameters carefully.

4 Experiments

We implemented a solver for the differential equation of (4) that is based on the Euler method with step size τ . After having reduced the equation to the first order, the variables that were updated at each t are q , \dot{q} , \ddot{q} , and $q^{(3)}$. The code and data we used to run the following experiments can be downloaded at <http://sailab.diism.unisi.it/motion-invariance/>, together with the full list of model parameters. We randomly selected two real world video sequences from the Hollywood Dataset HOHA2 [Marszałek *et al.*, 2009], that we will refer to as “skater” and “car”, and a clip from the movie “The Matrix” (©Warner Bros. Pictures). The frame rate of all the videos is ≈ 25 fps (we set $\tau = 1/25$), each frame was rescaled to 240×110 and converted to grayscale. Videos have different lengths, ranging from ≈ 10 to ≈ 40 seconds, and they were repeated in loop until 45,000 frames were generated, thus covering a significantly longer time span. We randomly initialized $q(0)$, while the derivatives at time $t = 0$ were set to 0. We used the softmax function to force a probabilistic activation of the features, and computed the optical flow v using an implementation from the OpenCV library. Convolutional filters cover squared areas of the input frame, and we set g_x to be the uniform distribution. All the results that we report are averaged over 10 different runs of the algorithms. The video is presented gradually to the agent so as to favour the acquisition of small chunks of information.

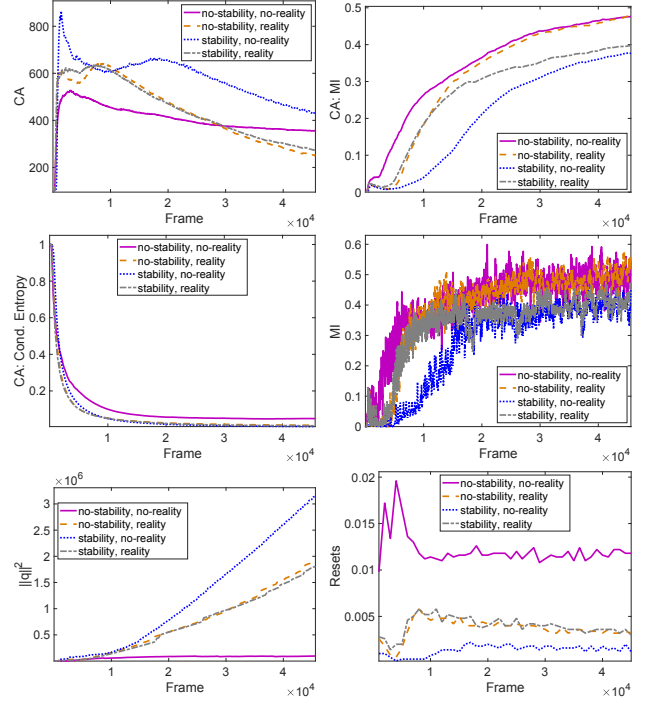


Figure 1: Comparing 4 configurations of the parameters, characterized by different properties in terms of stability and reality of the roots of the characteristic polynomial. The input video is reproduced (in loop) for 45k frames (x-axis). From left-to-right, top-to-bottom we report the Cognitive Action (CA), the portion of the cognitive action that is about the Mutual Information (MI) (that we maximize), the portion that is about the Conditional Entropy, the MI per-frame, the norm of $q(t)$, and the fraction of “reset” operations performed every 1000 frames.

In detail, $C(x, t) = \phi(t)[\text{gauss}(\delta(1 - \phi(t))) \ast C^o(x, t)]$, where \ast is the spatial convolution operator, $C^o(x, t)$ is the source video signal, $\text{gauss}(\sigma^2)$ is a Gaussian filter of variance σ^2 , and $\delta > 0$ is a customizable scaling factor. We start with $\phi(0) = 0$, and then ϕ is progressively increased as time passes, $\phi(t+1) = \phi(t) + \eta(1 - \phi(t))$ (we set $\eta = 0.0005$). We refer to the quantity $1 - \phi$ as the “blurring factor”. In order to be able to (approximately) satisfy the conditions in Eq. (5) we need to keep the derivatives small, so we implement a “reset plan” according to which the video signal undergoes a “reset” whenever the derivatives become too large. Formally, if $\|\dot{q}(t')\|^2 \geq \epsilon_1$, or $\|\ddot{q}(t')\|^2 \geq \epsilon_2$, or $\|q^{(3)}(t')\|^2 \geq \epsilon_3$ then we forced $\phi(t')$ to 0 ($\epsilon_j = 300 \cdot n$, for all j), and then we set to 0 all the derivatives.

Our experiments are designed (i) to evaluate the dynamics of the cognitive action in function of different *temporal regularities* imposed to the model weights (parsimony), and then (ii) to evaluate the *effects of motion*, that introduces a spatio-temporal regularization on single and multi-layer architectures. We recall that our learning task is fully unsupervised, so we focus on the transfer of information from each considered video stream to the learned features at different layers, evaluating the impact of the motion-based term. For this reason, we report the Mutual Information (MI) index, to-

Config	(a)	(b)	
	(Skater)	Blurring	($n = 10, 5 \times 5$)
$\bar{S}\bar{R}$	0.54 ± 0.07	Slow	0.35 ± 0.08
$\bar{S}R$	0.54 ± 0.08	Fast	0.39 ± 0.05
$S\bar{R}$	0.44 ± 0.11	None	0.34 ± 0.08
SR	0.45 ± 0.13		

Video	(c)	
	($n = 5, 5 \times 5$)	($n = 11, 11 \times 11$)
Car	0.38 ± 0.03	0.272 ± 0.003
Matrix	0.60 ± 0.03	0.45 ± 0.02
Skater	0.45 ± 0.13	0.35 ± 0.05

Table 1: MI on (a) the “skater” video, given the models of Fig. 1 (S =stability, R =reality, \bar{X} =not X); (b) different blurring plans (SR); (c) different videos, number of features n , filter sizes $\kappa \times \kappa$.

gether with other measurements on the internal state of systems.

Temporal regularities. When evaluating the temporal regularities, the cognitive action is composed by the entropy-based and parsimony terms only, and we experiment four instances of the set of parameters $\{\alpha, \beta, \gamma, k\}$. Each instance is characterized by the roots of the characteristic polynomial that lead to *stable* or *not-stable* configurations, and with only *real* or also *imaginary* parts, keeping the roots close to zero, and fulfilling the conditions of Eq. (7) when stability and reality are needed. These configurations are all based on values of $k \in [10^{-19}, 10^{-3}]$, while $\theta = 10^{-4}$. We performed experiments on the “skater” video clip, setting $n = 5$ features, and chose filters of size 5×5 . Results are reported in Fig. 1. The plots indicate that there is an initial oscillation that is due to the effects of the blurring factor, that vanish after about 10k frames. The Mutual Information (MI) (\mathcal{I}) portion of the cognitive action correctly increases over time, and it is pushed toward larger values in the two extreme cases of “no-stability, reality” and “no-stability, no-reality”. The latter shows more evident oscillations in the frame-by-frame MI value, due to the roots with imaginary part. The norm of q changes over time with different speeds, due to the small values of k , while the frequency of reset operations is larger in the “no-stability, no-reality” case, as expected. We evaluated the quality of the developed features by freezing the final q of Fig. 1 and computing the MI index over a single repetition of the whole video clip, reporting the results in Tab. 1 (a). This is the procedure we will follow in the rest of the paper when reporting numerical results in all the tables. We notice that, while in Fig. 1 we compute the MI on a frame-by-frame basis, here we compute it over the whole frames of the video at once, thus in a batch-mode setting. The result confirms that the two extreme configurations “no-stability, reality” and “no-stability, no-reality” show better results, on average. These performances are obtained thanks to the effect of the reset mechanism, that allows even such unstable configurations to develop good solutions. When the reset operations are disabled, we easily incurred into numerical errors due to strong oscillations, while for example, the “stability” cases were less

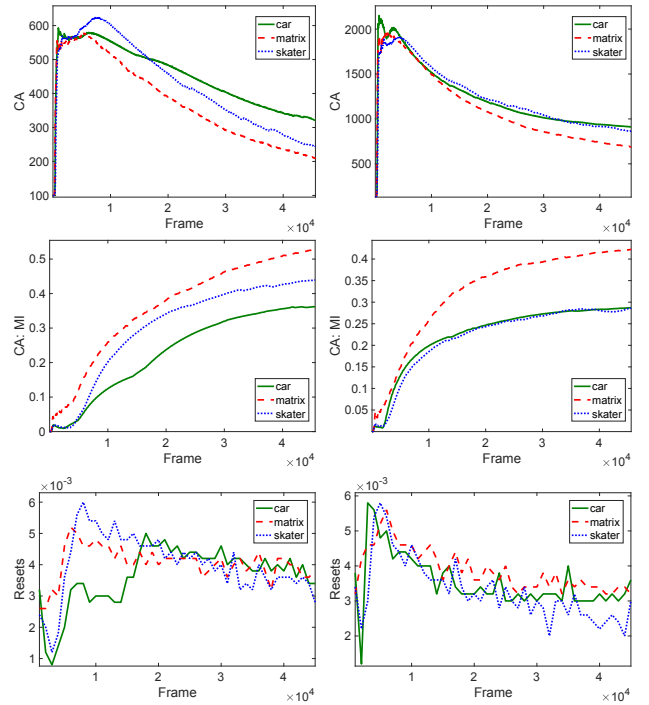


Figure 2: Different number of features and filter sizes (1st column: $n = 5, size = 5 \times 5$; 2nd column: $n = 11, size = 11 \times 11$) in 3 videos. See Fig. 1 for a description of the plots.

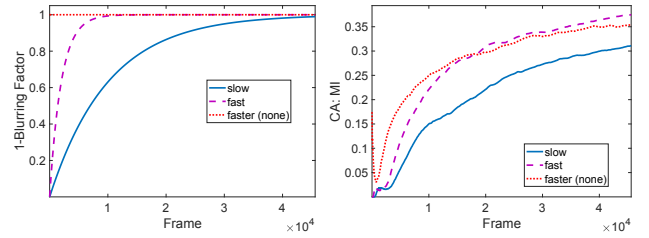


Figure 3: Three different blurring plans ($n = 11$ and filters of size 11×11).

affected by this phenomenon. We also compared the dynamics of the system on multiple video clips and using different filter sizes (5×5 and 11×11) and number of features ($n = 5$ and $n = 11$) in Fig 2. We selected the “stability, reality” configuration of Fig. 1, that fulfils the conditions of Eq. (7). Changing the video clip does not change the considerations we did so far, while increasing the filter size and number of features can lead to smaller MI index values, mostly due to the need of a better balancing the two entropy terms to cope with the larger number of features. The MI of Tab. 1 (c) confirms this point. Interestingly, the best results are obtained in the longer video clip (“The Matrix”) that requires less repetitions of the video, being closer to the real online setting. Figure 3 and Tab. 1 (b) show the results we obtain when using different blurring plans (“skater” clip), that is, different values of η that lead to the blurring factors reported in the first graph of Fig. 3. These results suggest that a gradual introduction of the video signal helps the system to find better

		$\lambda_M = 0$	10^{-8}	10^{-6}	10^{-4}	10^{-2}	1	10^2
Skater	$\ell = 1$.61 ± .11	.54 ± .11	.52 ± .07	.53 ± .08	.69 ± .07	.53 ± 0	.01 ± 0
	$\ell = 2$.53 ± .12	.62 ± .15	.60 ± .11	.43 ± .06	.48 ± .06	.1 ± .1	.03 ± .01
	$\ell = 3$.56 ± .17	.58 ± .20	.62 ± .10	.18 ± .16	.16 ± .17	.04 ± .02	.03 ± .02
Car	$\ell = 1$.49 ± .05	.44 ± .02	.46 ± .04	.47 ± .04	.66 ± .10	.60 ± .02	.01 ± 0
	$\ell = 2$.25 ± .26	.54 ± .10	.65 ± .08	.46 ± .03	.63 ± .11	.18 ± .32	.03 ± .01
	$\ell = 3$.26 ± .34	.45 ± .22	.51 ± .11	.38 ± .20	.24 ± .20	.09 ± .12	.04 ± .02
Matrix	$\ell = 1$.66 ± .01	.66 ± .02	.67 ± .01	.63 ± .05	.59 ± .03	.44 ± 0	.23 ± .02
	$\ell = 2$.55 ± .13	.56 ± .14	.43 ± 0	.45 ± .04	.62 ± .02	.35 ± .19	.13 ± .08
	$\ell = 3$.64 ± .03	.54 ± .11	.35 ± .07	.40 ± .01	.21 ± .07	.06 ± .03	.04 ± .02

 Table 2: MI in different videos, up to 3 layers ($\ell = 1, 2, 3$), and for multiple λ_M of the motion-based term. All layers share the same λ_M .

		$\lambda_M = 0$	10^{-8}	10^{-6}	10^{-4}	10^{-2}	1	10^2
Skater	$\ell = 2$.38 ± .34	.53 ± .12	.50 ± .1	.47 ± .1	.41 ± .02	.33 ± .17	.21 ± .2
	$\ell = 3$.55 ± .12	.62 ± .11	.55 ± .13	.42 ± .01	.36 ± .09	.2 ± .18	.39 ± .22
Car	$\ell = 2$.48 ± .1	.59 ± .17	.59 ± .18	.55 ± .12	.41 ± .01	.01 ± 0	.64 ± .01
	$\ell = 3$.67 ± .01	.60 ± .12	.73 ± .09	.36 ± .05	.33 ± .11	.27 ± .14	.73 ± .01
Matrix	$\ell = 2$.55 ± .13	.56 ± .14	.43 ± 0	.45 ± .04	.62 ± .02	.35 ± .19	.13 ± .08
	$\ell = 3$.55 ± .12	.53 ± .12	.82 ± .14	.35 ± .05	.35 ± .31	.02 ± .01	.01 ± 0

 Table 3: Same structure of Tab. 2. The model with the best λ_M is used as basis to activate a new layer (layer $\ell = 1$ is the same as Tab. 2).

solutions, but also that a too-slow process is not beneficial. The cognitive action has a big bump when no-plans are used, while this effect is more controlled and reduced in the case of both the slow and fast plans.

Effects of motion. In order to study the effect of motion in multi-layer architectures (up to 3 layers), we still kept the most stable configuration (“stability, reality”, 5×5 filters, 5 features), and introduced the motion-related term in the cognitive action. Our multi-layer architecture is composed of a stack of computational models developed accordingly to (3). A new layer ℓ is activated whenever layer $\ell - 1$ has processed a large number of frames ($\approx 45k$), and the parameters of layer $\ell - 1$ are not updated anymore. We initially considered the case in which all the layers $\ell = 1, \dots, 3$ share the same value λ_M that weighs the motion-based term. Tab. 2 shows the MI we get for different weighting schemes. Introducing motion helps in almost all the cases (for appropriate λ_M - the smallest values of λ_M are a good choice on average), and, as expected, a too strong enforcement of the motion-related term leads to degenerate solutions with small MI. We repeated these experiments also in a different setting. In detail, after having evaluated layer ℓ for all the values of λ_M , we selected the model with the largest MI and started evaluating layer $\ell + 1$ on top of it. Tab. 3 reports the outcome of this experience. We clearly see that motion plays an important role in increasing the average MI. In the case of “car”, we also obtained two (uncommon) positive results when strongly weighing λ_M . They are due to very frequent reset operations, that avoided the system to alter the filters when the motion-based term was leading to very large derivatives. This is an interesting behaviour that, however, was not common in the other cases we reported.

5 Conclusions

In this paper we have introduced a new approach to learning visual features according to the principle of least cognitive action. The experiments indicate the remarkable difference coming from the incorporation of motion invariance, with respect to the features only driven by information-based principles, which also results in the improvement of the mutual information from the video to the features. The theory is coherent with the different role of the ventral stream and dorsal stream [Goodale and Milner, 1992] that has been observed in humans. The enforcement of motion invariance is conceived for extracting features that are useful for object recognition to assolve the “what” task (ventral stream), whereas “dorsal neurons”, that are involved for where/how environmental interactions are expected not to use motion invariance. The model behind the learning of the filters indicates the need to access to velocity estimation, which is consistent with neuroanatomical evidence. Although the experimental results reported in the paper assume a uniform probability distribution in the spatial domain, the given formulation in the framework of the principle of least cognitive action suggests that the optimization must take place in areas of high saliency. In this case, the reformulation of the Euler equations given in this paper leads to identify the crucial role of eye movements in animals with foveal eyes. In future work we will also study the problem of building higher-level motion-based object predictors on top of the motion-invariant features described in this paper, with the same goal of giving a clear theoretical foundation to the development of such predictors.

References

- [Anderson and Rosenfeld, 1988] J.A. Anderson and E. Rosenfeld, editors. *Neurocomputing: Foundations of Research*. MIT Press, Cambridge, 1988.
- [Baker *et al.*, 2011] Simon Baker, Daniel Scharstein, J. P. Lewis, Stefan Roth, Michael J. Black, and Richard Szeliski. A database and evaluation methodology for optical flow. *Int. J. Comput. Vision*, 92(1):1–31, March 2011.
- [Betti and Gori, 2016] Alessandro Betti and Marco Gori. The principle of least cognitive action. *Theor. Comput. Sci.*, 633:83–99, 2016.
- [Deng *et al.*, 2009] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, pages 248–255, 2009.
- [Goodale and Milner, 1992] Melvyn A. Goodale and A. David. Milner. Separate visual pathways for perception and action. *Trends in Neurosciences*, 15(1):20–25, 1992.
- [Gori *et al.*, 2012] Marco Gori, Stefano Melacci, Marco Lippi, and Marco Maggini. Information theoretic learning for pixel-based visual agents. In *European Conference on Computer Vision*, pages 864–875. Springer, 2012.
- [Gori *et al.*, 2016] Marco Gori, Marco Lippi, Marco Maggini, and Stefano Melacci. Semantic video labeling by developmental visual agents. *Computer Vision and Image Understanding*, 146:9–26, 2016.
- [Horn and Schunck, 1981] B. K.P. Horn and B.G. Schunck. Determining optical flow. *Artificial Intelligence*, 17(1-3):185–203, 1981.
- [Huang *et al.*, 2007] Fu Jie Huang, Y-Lan Boureau, Yann LeCun, et al. Unsupervised learning of invariant feature hierarchies with applications to object recognition. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.
- [Kavukcuoglu *et al.*, 2010] Koray Kavukcuoglu, Pierre Sermanet, Y lan Boureau, Karol Gregor, Michael Mathieu, and Yann L. Cun. Learning convolutional feature hierarchies for visual recognition. In J. D. Lafferty, C. K. I. Williams, J. Shawe-Taylor, R. S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 1090–1098. Curran Associates, Inc., 2010.
- [Li *et al.*, 2016] Yin Li, Manohar Paluri, James M Rehg, and Piotr Dollár. Unsupervised learning of edges. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1619–1627, 2016.
- [Lowe, 2004] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [Marr, 1982] D. Marr. *Vision*. Freeman, San Francisco, 1982. Partially reprinted in [Anderson and Rosenfeld, 1988].
- [Marszałek *et al.*, 2009] Marcin Marszałek, Ivan Laptev, and Cordelia Schmid. Actions in context. In *IEEE Conference on Computer Vision & Pattern Recognition*, pages 2929–2936, 2009.
- [Melacci and Gori, 2012] Stefano Melacci and Marco Gori. Unsupervised learning by minimal entropy encoding. *IEEE Trans. Neural Netw. Learning Syst.*, 23(12):1849–1861, 2012.
- [Pathak *et al.*, 2017] Deepak Pathak, Ross B Girshick, Piotr Dollár, Trevor Darrell, and Bharath Hariharan. Learning features by watching objects move. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2701–2710, 2017.
- [Poggio and Anselmi, 2016] Tomaso A. Poggio and Fabio Anselmi. *Visual Cortex and Deep Networks: Learning Invariant Representations*. The MIT Press, 1st edition, 2016.
- [Sun *et al.*, 2014] Lin Sun, Kui Jia, Tsung-Han Chan, Yuqiang Fang, Gang Wang, and Shuicheng Yan. D1-sfa: deeply-learned slow feature analysis for action recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2625–2632, 2014.
- [Wang and Gupta, 2015] Xiaolong Wang and Abhinav Gupta. Unsupervised learning of visual representations using videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2794–2802, 2015.
- [Wiskott and Sejnowski, 2002] Laurenz Wiskott and Terrence J Sejnowski. Slow feature analysis: Unsupervised learning of invariances. *Neural computation*, 14(4):715–770, 2002.