ChaboNet : Design of a deep CNN for prediction of visual saliency in natural video

Souad Chaabouni, Jenny Benois-Pineau, Chokri Ben Amar

Elsevier Editorial System(tm) for Journal of Visual Communication and Image Representation
Manuscript Draft

Manuscript Number: JVCI-18-30R2

Title: ChaboNet : Design of a Deep CNN for Prediction of Visual Saliency in Natural Video

Article Type: Research paper

Keywords: Visual attention prediction in video; Deep Convolutional Neural Nteworks; saliency map; residual motion; dynamic content

Corresponding Author: Dr. souad chaabouni, Ph.D

Corresponding Author's Institution: Laboratoire bordelais d'informatique

First Author: souad chaabouni, Ph.D

Order of Authors: souad chaabouni, Ph.D; Jenny Benois-Pineau, Professor; Chokri Ben Amar, Professor

Abstract: Prediction of visual saliency in images and video is needed for video understanding, coding and other applications.
The majority of prediction models are founded only on "bottom-up" features.
Nevertheless, the "top-down" component of human visual attention becomes prevalent as human observers explore the visual scene. Visual saliency which is always a mix of bottom-up and top-down cues can be predicted on the basis of
seen data. In this paper, a model of prediction of visual saliency in video on the basis of Deep convolutional neural networks is proposed. A Deep CNN architecture is designed. Various input channels for a CNN architecture
are studied: using the known sensitivity of human visual system to residual motion, pixel colour values are completed with residual motion map. The experiments show that the choice of the input features for the Deep CNN depends on visual task.

Souad CHAABOUNI
LaBRI UMR 5800
351 crs de la Liberation F33405, France
Jenny BENOI
LaBRI UMR 5800
351 crs de la Liberation F33405, France
Chokri BEN AMAR
REGIM-Lab LR11ES48 ENIS BP1173, 3038 Sfax, Tunisie

12/01/2018
Dear ,

We wish to submit a new manuscript entitled "Chabonet: Design of a Deep CNN for Prediction of visual saliency in Natural Video" for consideration by the Journal of Visual Communication and Image Representation .We confirm tat this work is original. Some preliminary results were published in conference proceedings which we reference. The paper is not currently under consideration for publication elsewhere.
In this paper, we report on prediction of visual saliency using deep convolution network in natural video. The paper should be of interest to readers specialized in computer vision, visual saliency prediction and Deep learning.The novelty of the work is in the use of specific input layers in video, a transfer learning scheme and a good benchmark with classical saliency prediction models widely used by the community. Please address all correspondence concerning this manuscript tony self at :
chaabounisouad@yahoo.fr
Thank you for your consideration of this manuscript.

Yours Sincerely,
Souad CHAABOUNI
on behalf of all co-authors

We would like to thank the reviewers for their time spent evaluating our work and for their interest remarks which help us to better expose our paper.

Reviewer #2: Thank you for revising the paper by considering our comments. I think that the revised manuscript is better than the previous submission. However, there still exist several other major concerns.	
1 The authors claim that their first contribution is that the architecture is benchmarked against known architectures AlexNet and LeNet. I think this cannot be considered as a novel contribution. LeNet is proposed in 1994, and AlexNet is in 2012. Nowadays many architectures (e.g. VGG, ResNet) outperforms these two.	We have proposed a specific architecture in a family of « light » deep architectures such as AlexNet. Its particularities consist in the re-enforcement of features, as human visual system is sensitive to contrasts. We stress that we remain in the framework of such a «light » strategy in the section 3, see : « In this section, the proposed architecture ChaboNet for the visual saliency prediction is presented. This is a relatively "light" deep architecture compared to havier popular GoogleNet … » As our experiences show, the same architecture is applicable to very different datasets, subject to an adequate training and initialization, see section 4 « transfer learning ». Obviously, the architecture can be adapted to the new optimization strategies such as proposed in ResNet.
2 Fonts in Table 11, 12, 13 and 14 are too small.	The tables 11, 12, 13 and 14 are presented vertically, please see the page 31 .. 35
3 There are some typos in this paper, a thorough proofreading is needed. Here we list some examples in the following sentences. (1) On Page 5, in line 121: to avfoid overfitting -> to avoid overfitting; (2) On Page 26, in Table 9: %78.73+-0.930 -> 78.73%+-0.930	Typos corrected

ChaboNet : Design of a Deep CNN for Prediction of Visual Saliency in Natural Video

Souad CHAABOUNI[a,b], Jenny BENOIS-PINEAU[a], Chokri BEN AMAR[b]

[a] *LaBRI UMR 5800, Univ. Bordeaux, 33400 France*
[b] *Department of Computer science, REGIM-Lab LR11ES48, Univ. Sfax, 3038 Tunisia*

Abstract

Prediction of visual saliency in images and video is needed for video understanding, search and retrieval, coding, watermarking and other applications. The majority of prediction models are founded only on "bottom-up" features. Nevertheless, the "top-down" component of human visual attention becomes prevalent as human observers explore the visual scene. Visual saliency which is always a mix of bottom-up and top-down cues can be predicted on the basis of seen data. In this paper, a model of prediction of visual saliency in video on the basis of Deep convolutional neural networks (CNNs) is proposed. A Deep CNN architecture is designed. Various input channels for a CNN architecture are studied: using the known sensitivity of human visual system to residual motion, pixel colour values are completed with residual motion map. The latter is a normalized energy of residual motion in video frames with regard to the estimated global affine motion model. The experiments show that the choice of the input features for the Deep CNN depends on visual task: for highly dynamic content, the proposed model with residual motion is more efficient and gives decent results with relatively shallow Deep architecture.

Keywords: Visual attention prediction in video, Deep Convolutional Neural Networks, saliency map, residual motion, dynamic content.

URL: souad.chaabouni@u-bordeaux.fr (Souad CHAABOUNI), jenny.benois-pineau@u-bordeaux.fr (Jenny BENOIS-PINEAU), chokri.benamar@ieee.org (Chokri BEN AMAR)

1. Introduction and related work

Prediction of visual saliency in images and video is an intensively researched topic. With the growing volumes of digital video it is of highest interest for variety of applications involving video understanding, coding, watermarking... Prediction of pixel-wise saliency means assigning to each pixel in the image plane a measure characterizing the attraction of this image locus for a human observer. Several visual low-level characteristics: luminance, color, orientation and movement provoke human gaze attraction when observing visual content. This is why a very large amount of research works was devoted to prediction of visual saliency in images and video on the basis of popular "feature integration theory" [1]. These models simulate stimuli-driven, or "bottom-up" attention. [2] proposed a saliency detection algorithm for panoramic landscape images of outdoor scenes. Hence, the background of a panoramic image was estimated using the characteristics of geodesic similarity on a graph and the spatial distribution of homogeneous background regions. In [3], the saliency was considered as a ranking loss function which is designed to rank saliency values in the descending order of their relevance. In [4], the similarity computed between the input image and each similar image is measured and used for computing adaptive fusion weights for multiple saliency maps fusion.

For saliency detection in videos, local features and global features were extracted in [5] to create a pixel-level temporal and spatial saliency map. The authors of [6] defined the primary salient object in a video using the integration of the local visual/motion saliency, the global appearance consistency, and spatio-temporal smoothness constraint on object trajectories. In [7] the properties of saliency, low-rank, connectivity and sparsity were integrated into an unified objective function to detect a moving object using the saliency map. Models that use motion features are reported in [8]. The model of Wang [9] computes the gradient flow field and energy optimization using intra-frame boundary information and the inter-frame information to build a consistent spatio-temporal

saliency. In [10], a video saliency model was proposed in order to detect the attended regions. The latter correspond to both interesting objects and dominant motions. The so-called "top-down" attention is driven by observation task. In a task-driven visual search users search for particular objects and the goal of saliency models is to predict them in the image plane such as in [11]. In a free observation process of unknown video content, top-down visual attention is triggered when the observer understands the scene and selects the targets to follow. It becomes prevalent [12] when the human subject observes visual content with progressively increasing observation time, which is the case in continuous video scenes [13], [14]. Therefore, the areas of interest for the user cannot be predicted sufficiently well by purely bottom-up models. Various new cues have been proposed to enhance bottom-up models from the "top-down" perspective [12]. Boujut et al. [15] proposed a bottom-up spatio-temporal saliency model, enhanced by considering "face detection" as semantics in the video. The ELD [16] deep saliency model extracts high level features, using the deep network VGG-net, and low level features (Average RGB value, Gabor filter, ...). The concatenation of both encoded features, using a fully connected neural network, generates the final saliency map. The above mentioned research directly or not introduces elements of top-down visual attention prediction in model building. From methodological point of view frequent are the attempts to learn the interest of users with regard to elements of visual scenes whatever the content is when deploying supervised machine learning approaches [17]. Using supervised learning in the field of saliency prediction generates classifiers that can predict the focus of attention on the basis of already seen data thus combining bottom-up and top-down visual cues. Here, deep learning has emerged as an active research trend. It involves learning, at multiple levels of abstraction, for mining data such as images, sound, and text [18]. In addition to the common design of the classifier architectures on the basis of neural networks, deep learning presents a philosophy to model the complex relationships between data [19], [20]. Generally, deep neural networks are multi-layer predictive networks formed to maximize the probability of input data with regard to target classes

[21]. Due to the increased computational capacities with Graphical Processing Units (GPU) deep neural networks have outperformed all previous classification models in the tasks of video understanding. Deep convolutional neural networks (CNNs) were developed in Computer Vision, first by Yann LeCun with the LeNet [19] architecture that was used to recognize digits. Then, AlexNet [22] network has become very popular as the architecture for visual recognition tasks. Now, deep learning architectures, which have recently been proposed for the prediction of salient areas in images, differ essentially by the quantity of convolution and pooling layers, the input data, pooling strategies, the nature of the final classifiers, the loss functions to optimize and the formulation of the problem. Recently a fully convolutional network "FCN" for saliency prediction was proposed. This model estimates a dense saliency map of a given image using a set of extreme learners, each trained on an image similar to the input image [23]. A lot of works today, are devoted to saliency prediction in still images using "FCN": In [24], a global scene information that was trained on diverse categories of an eye-tracking data set, was used in addition to local information. [25] present an architectural extension to any CNN to fine-tune traditional 2D saliency prediction to Omni-directional Images. In [26], the authors proposed a deep CNN that predicts eye fixations and segments salient objects. [27] reuses an existing neural network trained on the task of object recognition to predict eye fixations. [28] formulated the prediction of eye fixations as a minimization of a loss function that measures the Euclidean distance of the predicted saliency map with the provided ground truth. Despite the popularity of these models, they still need a thorough study in real-life situations.

Prediction of visual attention in images reveals the binary classification problem for areas in images as "salient" and "non-salient". It corresponds to the visual experiment in free viewing conditions, when the subjects are simply asked to look at the content without any specific visual task. In [29], firstly, the learning of the relevant characteristics of the saliency of natural images was performed, and secondly the eye fixations on objects with semantic content was predicted. In Simonyan's work [30], the subjects are asked to look for an object from a

given taxonomy in the images. Therefore, the classification problem is multi-class, and can be expressed as a task-dependent visual experiment. In [31], first, a random bank of uniform filters is used to generate multiple representations of input images. The second phase provides the combination of different localized representations. While a significant effort has been already made for building saliency prediction models from still images with deep learning approach, very few have been built for video content with it [32]. Video has a supplementary dimension: the temporality that is expressed by apparent motion in the image plane. In such early works several combinations of primary features (input) such as color values and residual motion [33] are used to feed a CNN.

The intelligent application of Deep NNs to the well-studied problem of prediction of visual saliency in images and video requires deep understanding of their inherent weaknesses. The first one is the need of a very large amount of training data for building a well-generalizing model. Next, they are sensitive to the noise in training data. The noise is quite common when automatic and manual annotation of a large amount of data is required.

In the present paper we are interested in the prediction of saliency maps, which means the degree of interest for the observer in each pixel position. We do not consider the problem of a "scanpath" prediction [34], that is dynamics of gaze saccades across image plane. Hence, the Deep CNNs turn to be a right tool for such a map prediction. In this study, an approach with Deep CNN that ensures learning salient areas in order to predict pixel-wise saliency maps in video is proposed. We systematize our early works [32] and go further in studying our network architecture for the saliency detection problem. A specific attention is payed to the input layer of the proposed architecture, i.e. the data we extract form video to feed the network.

We stress that we remain with relatively "light" deep architecture, comparable to AlexNet. This is why it is benchmarked against known "light" architectures AlexNet and LeNet. This is our first contribution. The second contribution consists in the design of a transfer learning scheme to avoid overfitting on real-life small saliency data sets. It differs from usual approaches of

transfer learning in Deep CNNs classifiers, where the majority of researches use the pre-rained models on ImageNet whatever the target domain and classification problem is accordingly to the method first proposed in [21]. The method is benchmarked with Bengio's method [21].

Despite the appearance of fully-convolutional (FC) deep networks for saliency prediction [35] which allow prediction of dense saliency maps, our method remains "sparse". This means that we i)predict saliency of regularly sampled patches in video frames with a classical Convolutional Network architecture, ii)and then densify the map by interpolation. Both FC-nets and our scheme require interpolation. Nevertheless, we consider that in our case, it is easier to select training data which will not be corrupted by distractors and changes of the focus-of-attention along the time in video. We explain this choice in section 2.

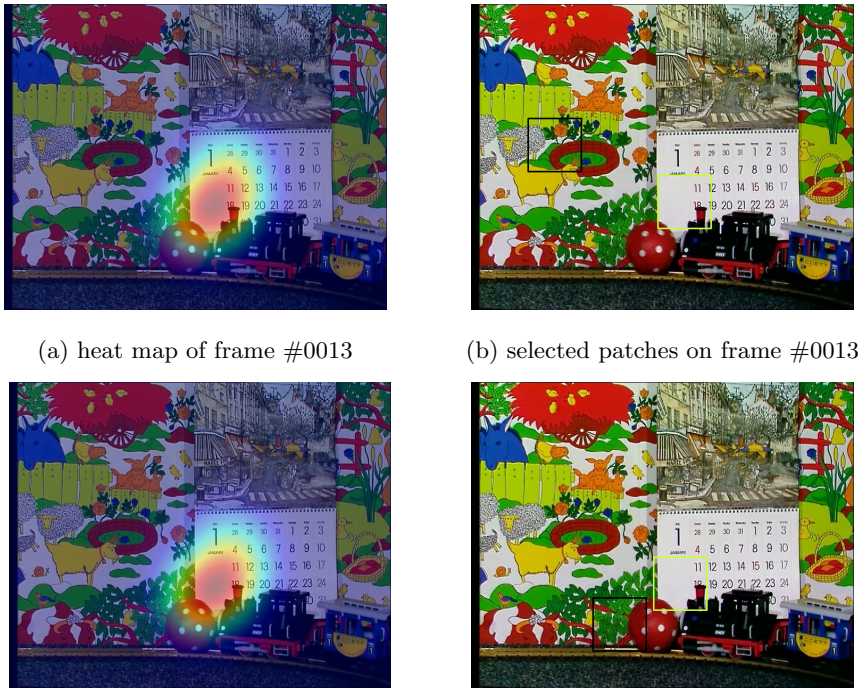
The paper is organized as follows. In section 2 data selection method is described, in section 3 the designed CNN architecture is presented, section 4 is devoted to the proposed transfer-learning scheme. Pixel-wise computation of predicted visual attention/saliency maps is then introduced in section 5. In section 6, results and comparison with classical saliency prediction methods are presented. Section 7 concludes the paper and outlines its perspectives.

2. Policy of data set creation: salient and non-salient patches

To train any CNN model selection of a training data set which would contain as less noise as possible is the must. At the first step of our method : prediction of saliency of patches in the video frames, classification problem is two-class. The training set has to comprise salient and Non-salient regions in video frames. The ground-truth for saliency here are the Gaze Fixation Density Maps (GFDM) [36]. They are built upon gaze fixations of a cohort of subjects recorded during a psycho-visual experiment.

For salient patches extraction the intuition is clear: we need to extract patches in the video frames where the GFDM has strong values. For Non-salient

patches extraction,the situation is more complex. First, due to the distractors and visual fatigue, the areas in a given video frame which are salient can become Non-salient in the next frame. Next, in the bottom-up saliency mechanisms of human visual attention, local contrasts can invoke human gaze. But if the observers are attracted by a semantic object in a different locus of a video frame, contrasted areas can have low values of GFDM. They thus become Non-salient. Hence, if the selection of Non-salient patches is based only on low GFDM values, then the resulting training data in Non-salient class of patches would contain noise. We illustrate such a phenomena in figure 1 (c). The focus of attention of subjects changes and Non-salient patches are selected even on the moving object (red ball). (a "Non-saillent" patch in figure 1 (b) is selected on a contrasted background).



(a) heat map of frame #0013

(b) selected patches on frame #0013

(c) heat map of frame #0014

(d) selected patches on frame #0014

Figure 1: Extraction of Non-salient patches by random selection in the Non-salient area of a video frame (SRC07 video IRCCyN [37]).

We thus proposed a strategy based on video production rules which will allows to reduce the noise in Non-salient training data. The method of patch selection was explained in detail in [38].Here we shortly remind it.

Figure 2 summarizes different steps to select salient patches. Firstly, the GFDMs are computed, then morphological erosion is applied. The illustration is given at a frame from HOLLYWOOD [1] data set. Patches centered on local maxima of saliency values are selected as salient. A Non-salient patch is a squared region in the image plane which is not supposed to attract human gaze. It should not be situated in the area-of-interest in a video frame, and must not be already selected as salient. According to the rule of thirds in produced and post-produced content, the most interesting details of the image or of a video frame have to cover the frame center and the intersections of the three horizontal and vertical lines that divide the image into nine equal parts [39]. Hence, we exclude the area -of-interest defined in the rule of thirds and selected salient patches when randomly selecting Non-salient patches in each video frame of training set.

3. Deep Convolutional Neural Network for visual saliency: ChaboNet

In this section, the proposed architecture ChaboNet for the visual saliency prediction is presented. This is a relatively "light" deep architecture compared to the popular GoogleNet [40], VGG [41] or variants of ResNet [42]. As the purpose is in predicting visual saliency in video and not in static images, specific features which are added to conventional RGB pixel values are described first. The implementation of ChaboNet is realized on the basis of Caffe framework [43].

3.1. A specific input data layer

When addressing visual attention prediction in video, the sensitivity of HVS to motion has to be taken into account [44]. The sensitivity of HSV to mo-

[1] available at http://www.di.ens.fr/∼laptev/actions/hollywood2/

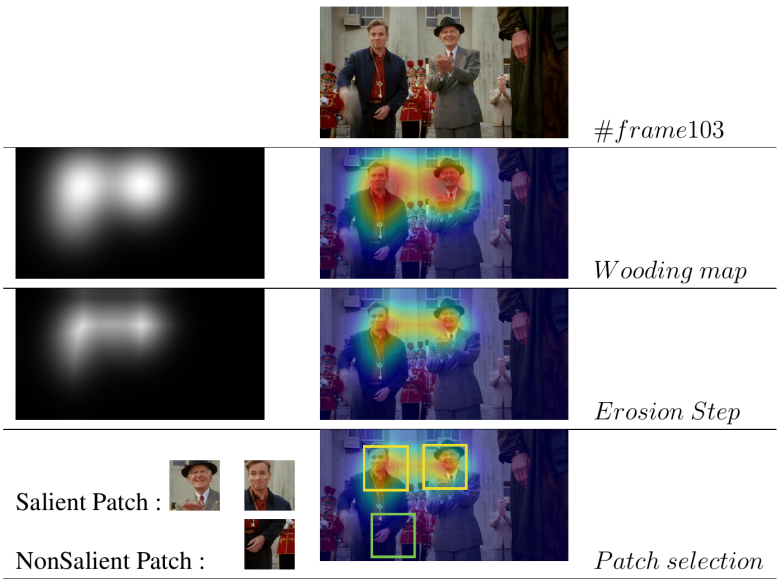


Figure 2: Policy of patch selection : example and steps, HOLLYWOOD data set

tion in a dynamic scene is modeled by residual motion [45]. Human observers accommodate to the global, i.e. ego-motion of the observer himself or camera motion in a recorded visual scene, and are attracted by specific local motions of objects. They first execute a saccade to a moving target and then continue with the "smooth pursuit" or visual tracking [46] keeping focus-of-attention on it. Local motion, of the target is expressed by residual motion relatively to the global camera motion [45]. To compute residual motion, the approach described in detail in [38] was followed. Here a pixel-wise motion field is computed by an optical flow method first. Using the dense motion field vectors as raw measures, the affine linear model of global motion is estimated by least square estimator and RANSAC algorithm [47]. Finally, the residual motion is the vector - difference between the initial motion vector and the one generated by the estimated affine model. As motion features, the squared $L2$ norm of residual motion vectors in each pixel in a video frame, normalized by its maximum in the frame, is used.

The composition of the input layer of the CNN is illustrated in figure 3. Here for each patch the input layer is composed of three color channel values and the residual motion feature map. Due to this configuration, the model is called "4K-model" in contrast to "3K-model", where only color channel values are used.

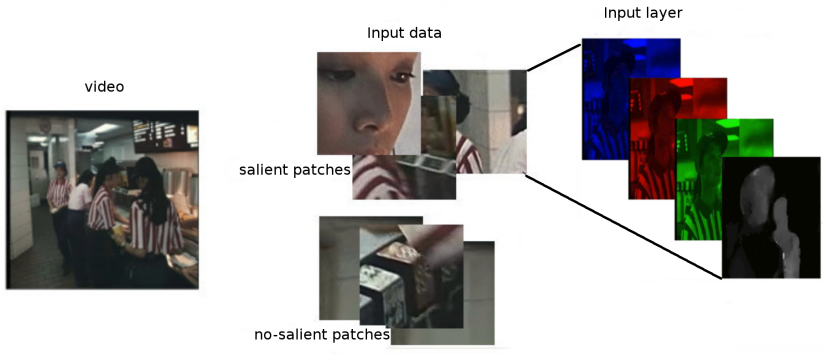


Figure 3: Input data layer : different features joined to the network.

3.2. The ChaboNet network architecture design

ChaboNet architecture was designed for the two-class classification problem: prediction of category of a patch in a given video frame as salient or non-salient. We aimed i) to preserve a reasonable deepness and ii) to remain comparable in the number of layers with a quite efficient network Alexnet [22]. The proposed ChaboNet architecture is summarized in figure 4.

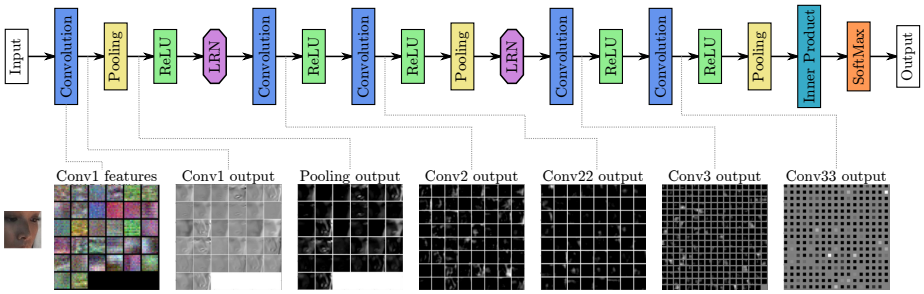


Figure 4: Architecture of video saliency convolution network 'ChaboNet'.

As in Deep CNN architectures designed for image classification tasks [43], ChaboNet is composed of a hierarchy of patterns. Each pattern consists of a cascade of operations, followed by a normalization operation in some cases. The result of normalization operation is denoted as $Norm$. The cascading of linear and nonlinear operations successively produces high-level features. They are transmitted via a fully connected layer to the deepest layer which is a softmax classifier. It assigns the confidence for each patch to be salient or not. Due to quite a limited size of input patches three patterns are proposed in this architecture. The pattern P^1 below is a usual combination of convolution, pooling and non-linear layers, P^2 and P^3 have the same structure. The whole network can be detailed as follows.

$$
\begin{aligned}
&Pattern\ P^1: \\
&\qquad Input \xrightarrow{convolution} Conv^1 \xrightarrow{pooling} Pool^1 \xrightarrow{ReLU} R^1 \\
&Pattern\ P^p:\ with\quad p \in \{2,3\} \\
&\qquad Norm^{p-1} \xrightarrow{convolution} Conv^p \xrightarrow{ReLU} R^p \xrightarrow{convolution} Conv^{pp} \xrightarrow{ReLU} R^{pp} \xrightarrow{pooling} Pool^p
\end{aligned}
$$

The normalization operation that leads to the output $Norm$, is added after the patterns P^1 and P^2 only, as after the pattern P^3 the features are quite sparse. The architecture of ChaboNet is depicted in figure 4. The features after convolution layers are presented for the example image from figure 3. It can be seen that the first layer of the network performs more as low-pass filters and deeper the convolution layer is more "high-pass" effect is observable.

3.2.1. Convolutional layers

In order to extract the most important information for further analysis or exploitation of image patches, the convolution with a fixed number of filters is needed. It is performed accordingly to the equation (1):

$$\mathbf{x}_j^l = \sum_{k \in \Omega_j} \mathbf{x}_k^{l-1} \odot \mathbf{w}_k^l + \mathbf{b}_j^l \tag{1}$$

with Ω_j - is the kernel support, i.e. the receptive field of j-th neuron;

l- is the network layer;

$\mathbf{x}_j^l$ - is the input of j-th neuron at layer l, that is feature-map vector;

$\mathbf{w}_k^l$ - is the weight of k-th neuron in the receptive field Ω_j;

$\mathbf{b}_j^l$ - is the bias of j-th neuron at the layer l.

$\odot$ is Hadamard product which is a coordinate-wise operation.

Inspired by literature as [22], [29] where the size of convolution kernels is either maintained constant or is decreasing with the depth of layers, in ChaboNet network, 32 kernels were used with the size of 12×12 for the convolution layer of the first pattern P^1. In the second pattern P^2, 128 kernels for each convolutional layer were used. In P^2 the size of the kernels for the first convolutional layer was chosen as 6×6 and for the second convolution layer, a kernel of 3×3 was used. Finally, 288 kernels with the size of 3×3 were used for each convolution layer of the last pattern P^3. This allows a progressive reduction of highly dimensional data before conveying them to the fully connected layers. The number of filters in the convolutional layers is growing, on the contrary, to explore the richness of the original data and to highlight structural patterns. For the filter size, several tests were made with the same values as in AlexNet [22], Shen's network [29], LeNet [19], Cifar [48] and finally, the size of 12×12 was retained in the first layer of the pattern P^1 as it yielded the best accuracy in saliency prediction problem. The bias term was chosen as a null term.

3.2.2. Pooling layers

Pooling reduces the computational complexity for the upper layers and summarizes the outputs of neighboring groups of neurons from the same kernel map. It reduces the size of each input feature map by the acquisition of a value for each receptive field of neurons of the next layer. In our architecture max-pooling was used. For each channel of data at the l-th network layer it is expressed by equation (2):

$$\mathbf{y}^l(x, y) = \max_{x', y' \in \mathcal{N}} \mathbf{x}^{l-1}(x', y\prime) \tag{2}$$

Here $\mathcal{N}$ denotes the neighborhood of (x, y). max operation is performed coordinate-wise. The kernel size of the pooling operation for the both patterns P^1 and P^2 was set to 3×3. The pooling of the third pattern P^3 was done with a size of 1×1, which means the full connection to the inner product layer.

3.2.3. Non Linear Response Layer

The non-linear transformation layer simulates the response of a neuron on excitement. In previous works on image classification the rectified linear unit (ReLU) function has been shown efficient [22]. The ReLU operation is expressed as (3)

$$\mathbf{z} = max(\mathbf{0}, \mathbf{x}) \tag{3}$$

It is also applied in a coordinate-wise manner

Compared to usual for neural networks sigmoid function, ReLU does not suppress high frequency features. This is a good property for saliency prediction task. Indeed, HVS is sensitive to contrasts. The first pattern P^1 is designed in the manner that the ReLU operation is introduced after the pooling one. As the operations of pooling and ReLU compute the maximum, they are commutative. Cascading pooling before ReLU can reduce the execution time as pooling step reduces the number of neurons or nodes. In the two last patterns, stacking two convolutional layers before the destructive pooling layer ensures the computation of more complex features that will be more "expressive".

3.2.4. Local response normalization layers

A local Response Normalization (LRN) layer normalizes values of feature maps which are calculated through the neurons having unbounded (due to ReLU) activation. This operation is used to detect the high-frequency characteristics with a high response of the neuron, and to scale down answers that are uniformly greater in a local area (see equation (4)).

$$\psi(\mathbf{z}(x,y)) = \frac{\mathbf{z}(x,y)}{(1+\frac{\alpha}{N^2}\sum_{x'=\max(0,x-[N/2])}^{\min(S,x-[N/2]+N)}\sum_{y'=\max(0,y-[N/2])}^{\min(S,y-[N/2]+N)}(\mathbf{z}(x',y'))^2)^{\beta}} \tag{4}$$

Here $\mathbf{z}(x, y)$ represents the value of the feature map after ReLU operation at (x, y) coordinates and the sums are taken in the neighborhood of (x, y) of size $N \times N$, α and β regulate normalization strength. Normalization is also a coordinate-wise operation. Figure 5 summarizes the parameters used for each layer of the three patterns.

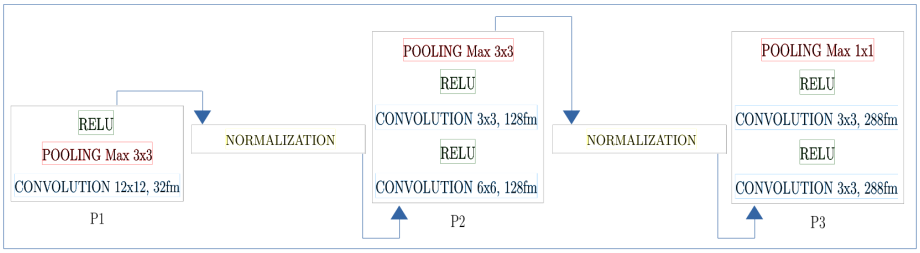


Figure 5: Detailed setting of each layer of 'ChaboNet' network.

3.3. Training and validation of the model

At a training step, the coefficients of convolution filters are repeatedly optimized in a forward-backward loop in the Caffe framework[43]. The optimization method used is the stochastic gradient descent 'SGD' with momentum. The initialization of convolution coefficients is realized randomly according to Gaussian law as proposed in [43]. The setting of the solver requires the definition of the number of iterations. It is defined accordingly to the equation (5):

$$iterations\_numbers = epochs \times \frac{Total\_images\_number}{batch\_size} \tag{5}$$

here *batch_size* represents the number of images for each network switching, *epochs* is the number of times the totality of the data set is switched by the network. The accuracy and loss are recorded with the interval of 2000 iterations both on training and validation sets.

4. Domain-dependent transfer learning for small databases

The generalization power of Deep CNN classifiers strongly depends on the quantity of data and on the coverage of data space in the training data set. In

real-life applications, e.g. prediction of benchmark models for studies of visual attention of specific populations [49] or saliency prediction for visual quality assessment [37] the database volumes are small. In order to predict saliency in these small collections of videos, transfer learning approach was needed. It presents a technique used in the field of machine learning that increases the accuracy of learning either by using it in different tasks, or in the same task [50]. Several studies have proven the power of this technique [51], [52]. In terms of optimization method which is SGD, transfer learning means that the network parameters are not initialized randomly, but their initialization corresponds to a local minimum of loss function for a large data set. A small database can be considered as slightly different data and starting from pre-trained parameter values can bring improvement in optimization.

In saliency prediction problem, one could have supposed that as HVS is sensitive to the singularities such as e.g. contrasts, the same trained saliency model can be applied whatever the database is. In [49] we have shown that it is not the case. Parameters trained on a large database for the same classification task are not efficient when directly applied to another database, we observe the so-called "over-fitting". Hence, the transfer learning or fine-tuning of the parameters from a large database to a small database is necessary.

Transfer learning scheme consists of two steps: i) learning the whole classification model on a large data set, ii) transfer on small data set. The latter means initialization of parameters' values in learning process by the optimal parameter values obtained on a large data set. In the present work, step i) - learning on a large data set was performed from a scratch, i.e. with random initialization of parameters at all layers. For the step ii) two initialization schemes were tested: that one proposed by Bengio et al. [50] and ours [33] explained in the following. For both steps i) and ii) the classification problem was the same : binary classification of patches into salient and non-salient.

In the research of Bengio et al. [50] addressing object recognition problem, the authors show that the first layers of a Deep CNN learn characteristics similar to the responses of Gabor filters regardless of the data set or classification

task. Hence, in their transfer learning scheme just the three first convolutional layers pre-trained on a large database are used for the other database as the initialization of parameters. The coefficients on deeper layers are left free for optimization, that is initialized randomly.

In our case, saliency prediction task differs from object recognition task. Thus the proposal is to initialize all parameters in all layers of the network to train on a small data set, by the best model trained on a large data set. Equation (6) expresses the transfer of the convolutional weights, $W^l = \{\mathbf{w}_k\}^l$ for layer l, obtained from the larger database to the new smaller database. Here the Stochastic Gradient Descent with momentum is used as in [43]. In the following equation we omit any indexes except iteration number i for simplicity:

$$\begin{cases} V_{i+1} = m \cdot V_i - \gamma \cdot \epsilon \cdot W_i - \epsilon \cdot \langle \frac{\partial L}{\partial W} | W_i \rangle_{D_i} \\ W_{i+1} = W_i + V_{i+1} \qquad | \qquad W_0 = W' \end{cases} \tag{6}$$

With $\epsilon = 0.001$- a fixed learning rate, $m = 0.9$ - momentum coefficient; $\gamma = 0.00004$ - weight decay and W' presents the best learned model parameters pre-trained on the large data set. The initial value of the velocity V_0 was set to zero. These parameter values are inspired by the values used in [43] with the same fixed learning rate and show the best performances on a large training data set. Hence in this section, the network for classification of patches in video frames as salient or non has been presented. In the next section, the method for generation of pixel-wise saliency maps on the basis of predicted salient patches is introduced.

5. Generation of saliency map

The saliency map of each frame I of the video is computed using the output value of the trained deep CNN model. Here we use the method from previous works [33], [38]. The soft-max classifier gives the probability for a patch of

belonging to the "salient" class accordingly to the equation 7.

$$\phi(\mathbf{u})_q = \frac{e^{u_q}}{\sum_r e^{u_r}}, r = 1, ..., d \tag{7}$$

To build a dense predicted saliency map we classify patches of the input video frames first. The patches of the same size s ($s = 100$ in our experiments) are sampled with a stride of 5 pixels. The output value of the soft-max classifier with regard to the salient class on each patch defines its degree of saliency. If the score is assigned to the center of each patch (x_0, y_0), a sparse saliency map is obtained $M(x, y)$. Then, to densify the map, score values are interpolated with Gaussian filters: in the center of each patch, a Gaussian $G(x, y)$ is applied with a peak value of $\frac{A*M(x_0,y_0)}{2\pi\sigma^2}$. The A-parameter value was experimentally chosen as 10. The spread parameter σ was fixed as a half-size of the patch. For each pixel in the image plane, the Gaussian are summed-up. Finally the map is normalized by saliency peak as in Wooding method for GFDM (see section 2).

6. Experiments and results

6.1. Data sets

To learn the model, three data sets were used, HOLLYWOOD[53] [54], the well-known CRCNS [55] and IRCCyN [37].

The HOLLYWOOD database contains 823 training videos and 884 videos for the validation step. The number of subjects with recorded gaze fixations varies according to each video with up to 19 subjects. The spatial resolution of videos varies as well. Despite the discrepancy of these parameters, we use it for model building as it is the only large-scale video database with recorded gaze fixations. The CRCNS [2] data set [55] is one of the oldest and the most known data sets for saliency prediction benchmarking. It contains 50 videos of 640×480 resolution Gaze recordings of up to eight different subjects are available. To create the training, validation and testing set, each video of CRCNS was split according to

[2] available at https://crcns.org/data-sets/eye/eye-1

the following scheme: one frame for testing, one frame for validation and four frames for training set were selected.

IRCCyN database is composed of 31 Standard-Definition (720x576) videos and comprises gaze fixations of 37 subjects. It is a homogeneous database designed for video quality assessment and by its good visual quality and the interest of the content, is the most interesting one. These videos contain certain categories of attention attractors such as high contrast, faces. However, videos with objects in motion are not frequent. Our model was designed for saliency prediction in video, to capture "smooth pursuit" motion of human eyes. It cannot be evaluated by using all available videos of IRCCyN data set. Hence, videos that do not contain a real object motion were eliminated. Only SRC02, SRC03, SRC04, SRC05, SRC06, SRC07, SRC10, SRC13, SRC17, SRC19, SRC23, SRC24 and SRC27 video files were thus used in experiments. This data set is referenced in the following as IRCCyN-MVT. For each chosen video of IRCCyN-MVT, the same selection scheme as for CRCNS data set was used: one frame is taken for the testing step, one frame for the validation step and four frames for the training step. The distribution of selected data between salient and non-salient classes is presented in table 1.

Table 1: Distribution of learning data: total number of salient and non-salient patches selected from each database.

data sets		training step	validation step
HOLLYWOOD	SalientPatch	222863	251294
	Non-SalientPatch	221868	250169
	total	444731	501463
CRCNS	SalientPatch	33370	8373
	Non-SalientPatch	30491	7730
	total	63861	16103
IRCCyN-MVT	SalientPatch	2013	511
	Non-SalientPatch	1985	506
	total	3998	1017

6.2. Evaluation of the interest of residual motion

To evaluate the interest of residual motion in saliency prediction, we have computed the AUC metric[34] between gaze fixations and the energy-of-residual-motion map. Here, we used popular data sets CRCNS [55] and IRCCyN [37] that have been created and benchmarked for the task of saliency prediction in natural videos.

Results summarized in table 2 and 4 show a correspondence between gaze fixations and residual motion map especially for the "gamecube02" video of CRCNS database with 0.56 value of AUC metric, and for the "SRC23" video of IRCCyN database where we obtain a very interesting result ($AUC = 0.68$). In table 4, 8 videos from 12 tested videos give an AUC value higher than 0.55. In remaining sequences "SRC02", "SRC07" and "SRC13' moving objects are not significant for their understanding. This experience can just encourage us to go for the integration of residual motion as an input to ChaboNet architecture.

Table 2: The comparison of AUC metric of gaze fixations 'GFM' vs the energy of residual motion map 'ResidualMotion' for 890 frames of CRCNS videos.

VideoName	$TotFrame = 890$	GFM vs ResidualMotion
beverly03	80	0.54 ± 0.119
gamecube02	303	0.56 ± 0.152
monica05	102	0.52 ± 0.110
standard02	86	0.499 ± 0.06
tv-announce01	73	0.472 ± 0.181
tv-news04	82	0.535 ± 0.186
tv-sports04	164	0.500 ± 0.147

Table 3: Frames of CRCNS videos.

beverly03	gamecube02	monica05	standard02	tv-announce01	tv-news04	tv-sports04

Table 4: The comparison of AUC metric of gaze fixations 'GFM' vs Residual Motion map for 456 frames of IRCCyN videos.

VideoName	$TotFrame = 456$	GFM vs ResidualMotion
SRC02	37	0.46 ± 0.025
SRC03	28	0.55 ± 0.112
SRC04	35	0.55 ± 0.191
SRC05	35	0.57 ± 0.148
SRC06	36	0.603 ± 0.156
SRC07	36	0.48 ± 0.028
SRC10	33	0.55 ± 0.086
SRC13	35	0.59 ± 0.147
SRC17	42	0.48 ± 0.071
SRC19	33	0.64 ± 0.078
SRC23	40	0.68 ± 0.094
SRC24	33	0.51 ± 0.045
SRC27	33	0.53 ± 0.074

6.3. Evaluation of patches' saliency prediction with ChaboNet

The network was implemented using a graphic card Tesla K40m and processor (2×14 cores). A sufficiently large amount of patches, 256, was used per iteration (see the *batch_size* parameter in equation (5)). After a fixed number of training iterations, a model validation step was implemented: here the accuracy of the model at the current iteration was computed on the validation data set. It is denoted as "Test accuracy" in the figures 6, 7, 8.

To evaluate the deep network and to prove the importance of the addition of the residual motion map, two models were created with the same parameter settings and architecture of the network: the first one contains R, G and B primary pixel values in patches, denoted as $ChaboNet3k$. The $ChaboNet4k$ is the model using RGB values and the normalized energy of residual motion as input data, see section 3.1.

Figure 6 illustrates the variations of the accuracy along iterations of all the models tested for the database HOLLYWOOD. Peak and mean accuracy values are presented in table 5). The results of learning experiments on HOLLYWOOD data set yield the following conclusions: i) when adding residual motion as an input feature to RGB values, the accuracy is improved by almost 2%. ii) the accuracy curve (figure 6 (a)) and the corresponding loss curve (figure 6(b)) show that the best trained model reached 80% of accuracy with the smallest

loss (at the iteration #8690 see table 5). Thus, it does not present an over-fitting situation. Figures 6 (c) and (d) show a better performance of the *ChaboNet4k* model in terms of speed for training and validation. Hence, the loss curve (Figure 6 (c)) of the *ChaboNet4k* model still decreasing during training. But since the 10000 iterations, the test loss curve (6 (d)) starts increasing which reflects the beginning of an over-fitting situation.

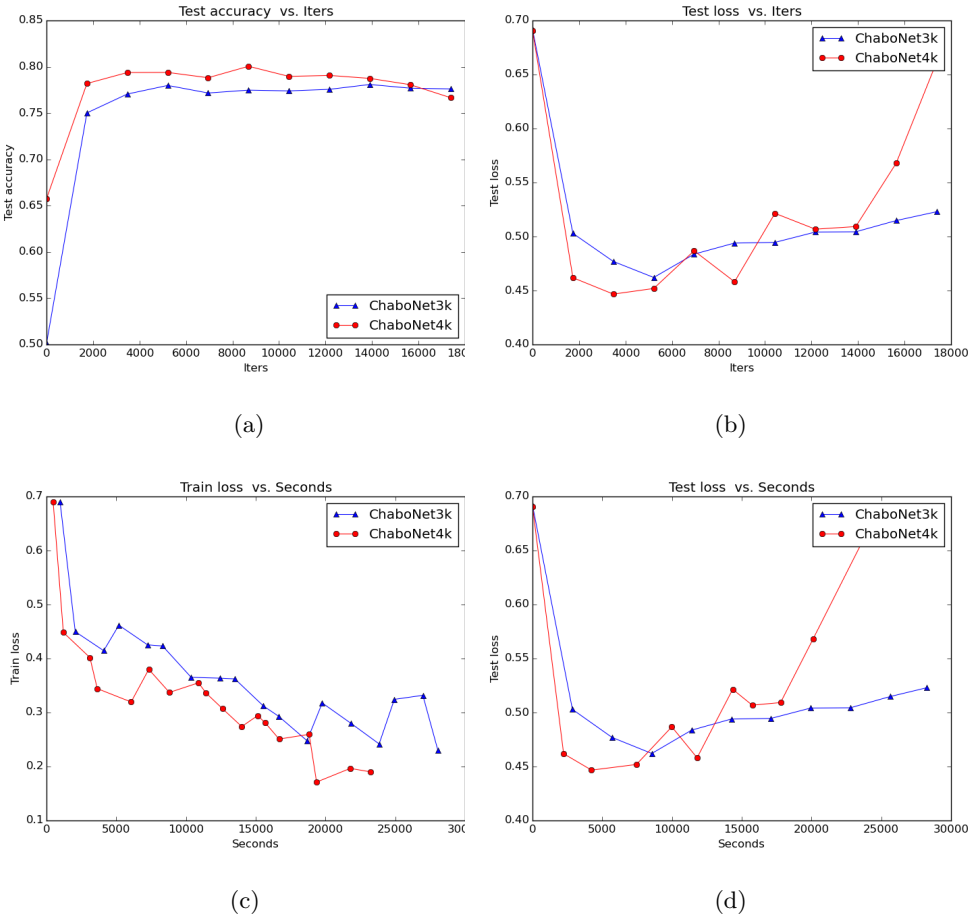


Figure 6: Training the network - Accuracy and loss vs time of *ChaboNet3k* and *ChaboNet4k* for HOLLYWOOD database : (a) Accuracy vs iterations, (b) Loss on validation data set vs iterations, (c) Train loss vs seconds, (d) Loss on validation data set vs seconds.

So, the model obtained after 8690 iterations is used to predict saliency on the validation set of this database, and to initialize the parameters when learning with transfer on other used data sets. Mean accuracy is also slightly higher. Indeed, 1.53% of mean accuracy increase is observed with merely the same stability of training which expressed by the standard deviation in the table 5.

Table 5: The accuracy results on HOLLYWOOD data set

	$ChaboNet3k$	$ChaboNet4k$
$training - time$	$7h47m33s$	$6h27m2s$
$min - Accuracy_{(\#iter)}$	$50.11\%_{(\#0)}$	$65.73\%_{(\#0)}$
$max - Accuracy_{(\#iter)}$	$77.98\%_{(\#5214)}$	$80.05\%_{(\#8690)}$
$avg - Accuracy \pm std$	$77.30\% \pm 0.864$	$78.73\% \pm 0.930$

Figure 7 illustrates the variations of the accuracy along iterations of all models tested on IRCCyN-MVT data set. To overcome the lack of data for training, the proposed transfer learning scheme, see section 4 was used from HOLLYWOOD to IRCCyN-MVT. The best HOLLYWOOD model obtained at the iteration 8690 was used to initialize training of parameters with $ChaboNet4k$ model on IRCCyN-MVT data set. For $ChaboNet3k$, training started from the iteration 5214 of the best model on HOLLYWOOD data set. Table 6 summarizes obtained results. The gain of using 4k- against 3k- data as input layer to the deep CNNs is about 1.12% in terms of mean accuracy.

Table 6: The accuracy results on IRCCyN-MVT data set.

	$ChaboNet3k$	$ChaboNet4k$
$training - time$	$0h4m6s$	$0h4m25s$
$min - Accuracy_{(\#iter)}$	$70.80\%_{(\#5216)}$	$77.83\%_{(\#8848)}$
$max - Accuracy_{(\#iter)}$	$92.67\%_{(\#6544)}$	$92.77\%_{(\#9664)}$
$avg - Accuracy \pm std$	89.96 ± 4.159	$91.08\% \pm 3.107$

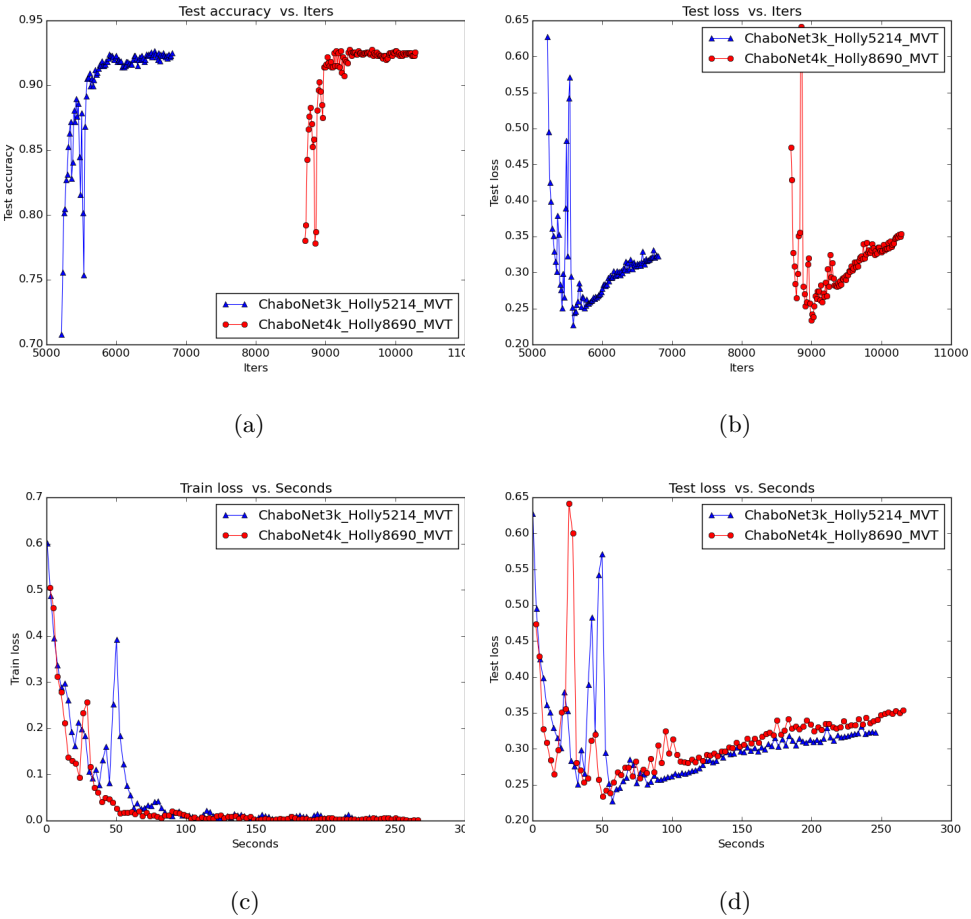


Figure 7: Accuracy and loss vs time of 3k and 4k for videos with motion from IRCCyN-MVT database : (a) Accuracy vs iterations, (b) Loss on validation data set vs iterations, (c) Train loss vs seconds, (d) Loss on validation data set vs seconds.

Figure 8 illustrates the variations of the accuracy and loss along the time expressed in number of iterations and seconds for CRCNS data set. The best model is obtained at the iteration #32500 with an accuracy of 91.66%.

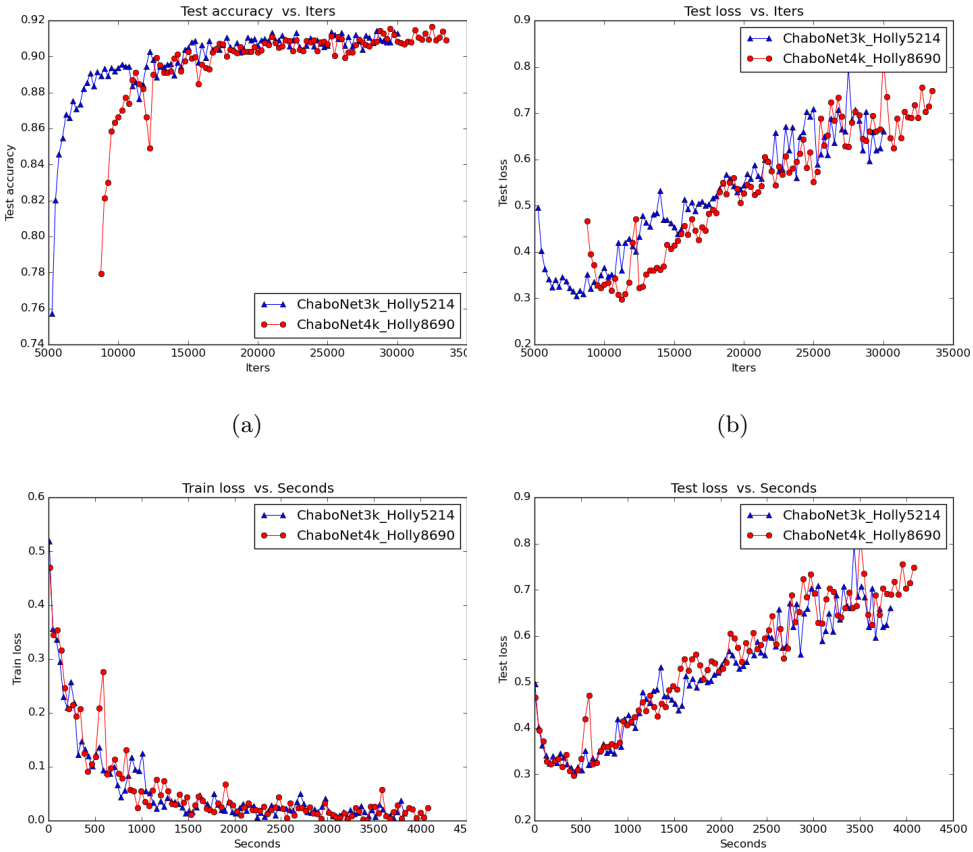


Figure 8: Accuracy and loss vs iterations of 3k and 4k for CRCNS database : a) Accuracy vs iterations, (b) Loss on validation data set vs iterations, (c) Train loss vs seconds, (d) Loss on validation data set vs seconds.

Table 7: The accuracy results on CRCNS data set

	$ChaboNet3k$	$ChaboNet4k$
$training - time$	$1h3min42s$	$1h7min58s$
$min - Accuracy_{(\#iter)}$	$75.71\%_{(\#5250)}$	$77.95\%_{(\#8750)}$
$max - Accuracy_{(\#iter)}$	$91.45\%_{(\#28500)}$	$91.66\%_{(\#32500)}$
$avg - Accuracy \pm std$	$89.77\% \pm 2.085$	$89.81\% \pm 2.035$

6.4. Validation of the ChaboNet architecture

To evaluate the ChaboNet *architecture* designed for saliency prediction, an experiment was conducted with the HOLLYWOOD data set. The popular AlexNet [22] and the original LeNet [19] network architectures were used as a base-line with data patches extracted from HOLLYWOOD data.

For AlexNet, the network settings were taken exactly as in [22], that means the same number and size of filters at all layers and the same learning parameter such as the number of iterations (450.000). To better visualize, in figure 9 the iterations of AlexNet were limited to 70.000. Similarly, the original settings of LeNet were preserved from [19]. Here the number of iterations was 10.000. ChaboNet Network training was performed with 17.400 iterations.

Obtained results summarized in figure 9 showed that the ChaboNet network outperformed the AlexNet and LeNet architectures (see table 8). In fact, with 17.400 iterations, ChaboNet outperformed by 2% in mean accuracy the AlexNet architecture which needs 450.000 iterations. When comparing the 10.000 first iterations of ChaboNet and LeNet, mean accuracy was discovered to be better by more than 20%. Furthermore, the stability of training expressed by small standard deviation is much stronger, see line 4 of the table 8.

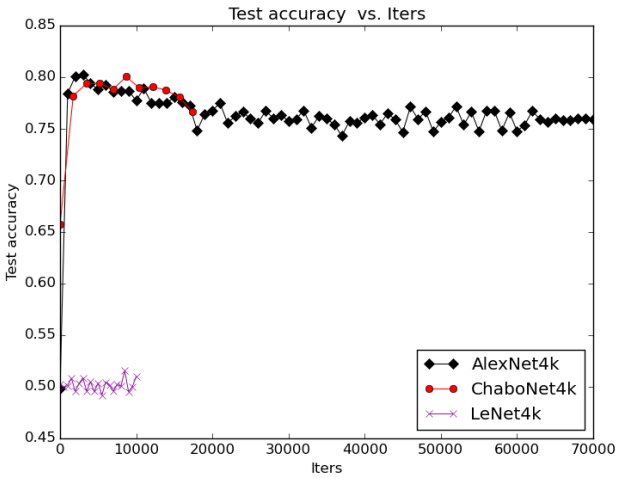


Figure 9: ChaboNet 4k architecture vs AlexNet and LeNet on HOLLYWOOD data set.

Table 8: Accuracy results : validation of ChaboNet 4k architecture vs AlexNet and LeNet networks on HOLLYWOOD data set.

	$ChaboNet4k$	$AlexNet4k$	$LeNet4k$
$min_{(\#iter)}$	$65.73\%_{(\#0)}$	$49,84\%_{(\#0)}$	$49,2\%_{(\#5500)}$
$max_{(\#iter)}$	$80.05\%_{(\#8690)}$	$80,27\%_{(\#3000)}$	$51,56\%_{(\#8500)}$
$avg \pm std$	$78.73\% \pm 0,930$	$76,77\% \pm 6,633$	$50,17\% \pm 0,575$

6.5. BN or LRN normalization for ChaboNet4k architecture

Recently, batch normalization 'BN' [56] which allows normalizing layer inputs, have shown its efficiency in designed architectures for image classification tasks. We compare and evaluate the use of batch normalization and the local response normalization for saliency prediction task. Figure 10 illustrates the variations of the accuracy and loss along the time expressed in number of iterations and seconds for the ChaboNet4k architecture using batch normalization 'BN' and local response normalization 'LRN'. Obtained results summarized in table 9, showed that the use of LRN layer outperforms the BN layer with 7% for saliency prediction tasks.

Table 9: The accuracy results on HOLLYWOOD data set

	$ChaboNet4k - LRN$	$ChaboNet4k - BN$
$min - Accuracy_{(\#iter)}$	$65.73\%_{(\#0)}$	$45.71\%_{(\#0)}$
$max - Accuracy_{(\#iter)}$	$80.05\%_{(\#8690)}$	$76.82\%_{(\#15642)}$
$avg - Accuracy \pm std$	$78.73\% \pm 0.930$	$70.78\% \pm 7.034$

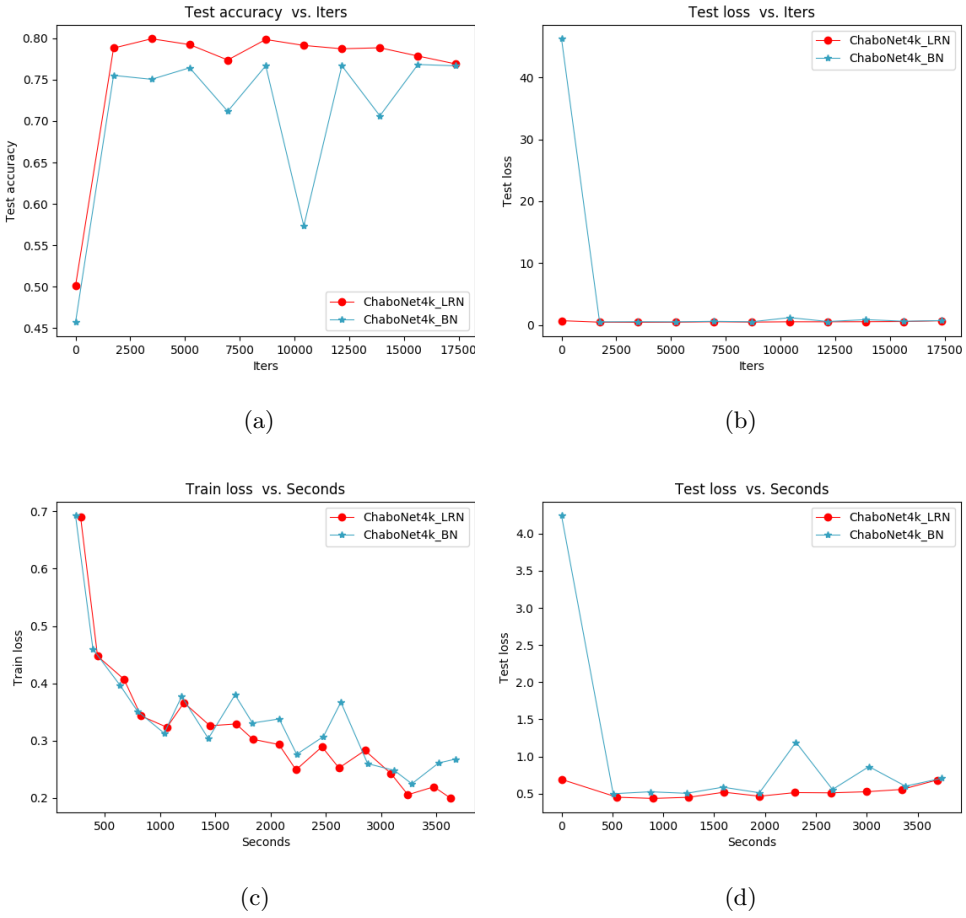


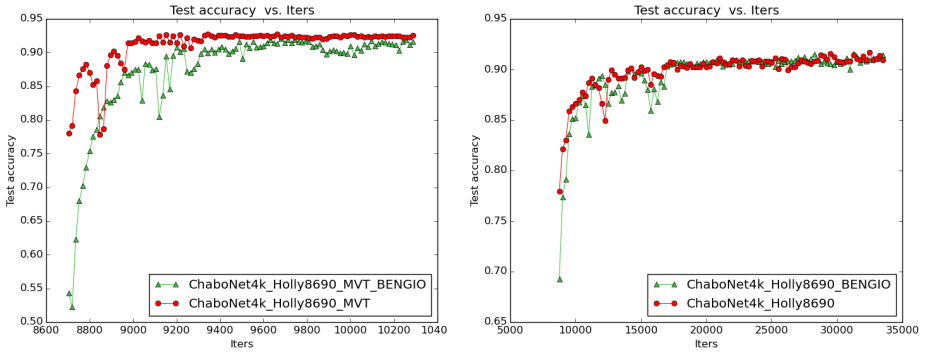
Figure 10: Training the network - Accuracy and loss vs time of *ChaboNet4k* for HOLLYWOOD database using local response normalization (LRN) and batch normalization (BN) : (a) Accuracy vs iterations, (b) Loss on validation data set vs iterations, (c) Train loss vs seconds, (d) Loss on validation data set vs seconds.

6.6. Validation of the proposed method of transfer learning

The previous work [33] already showed that training on a small database (IRCCyN) with transfer learning increases the accuracy by 4% in average and makes training process more stable (up to 50% of decrease of standard deviation of accuracy along iterations). In the present work, the proposed transfer

learning scheme is benchmarked with that one proposed by Bengio. Two experiments were conducted with the same small data set IRCCyN-MVT and CRCNS, and the same definition of network ChaboNet: i) Our method: start training of all ChaboNet layers from the best model already trained on the large HOLLYWOOD data set (see section 4). ii) Bengio's method: the three first convolutional layers are trained on the HOLLYWOOD data set and then fine-tuned on the target data set, other layers are trained on target data set with random initialization.

Figure 11 illustrates the variations of the accuracy along iterations of the two experiments performed with the two small data sets. One can see less stable behavior when the transfer method of Bengio et al. is applied. The proposed method of transfer learning outperformed the Bengio's method by almost 3.6% in IRCCyN-MVT data set and almost 0.44% in CRCNS data set in terms of mean accuracy. The gain on stability of training in our method is more than 50%, see table 10.



(a) (b)

Figure 11: Evaluation and comparison of the proposed method of transfer learning : (a)Comparison on IRCCyN-MVT, (b) Comparison on CRCNS.

Table 10: The accuracy results on IRCCyN-MVT and CRCNS data set.

	Our transfer method		*BENGIO transfer method*	
	IRCCyN-MVT	CRCNS	IRCCyN-MVT	CRCNS
$max_{(\#iter)}$	$92.77\%_{(\#9664)}$	$91.66\%_{(\#32500)}$	$92.08\%_{(\#9680)}$	$91.55\%_{(\#31250)}$
$avg \pm std$	$91.08\% \pm 3.107$	$89.81\% \pm 2.035$	$87.48\% \pm 7.243$	$89.37\% \pm 3.099$

6.7. Evaluation of predicted visual saliency maps

After training and validation of the model on HOLLYWOOD data set, the model obtained at the iteration #8690 having the maximum value of accuracy 80.05% was chosen. This model was used to predict the probability of a patch to be salient. For the CRCNS data set, the model obtained at the iteration #32500 with the accuracy of 91.66% is used to predict saliency. In the same manner, the model with the accuracy of 92.77% obtained at the iteration #9664 is used for the IRCCyN-MVT data set.

To evaluate the method of saliency prediction by interpolation of classifier outputs as presented in section 5, performances were compared with the most popular saliency models from the literature. Several spatial saliency models were chosen: Itti and Koch spatial model [57], Signature Sal [58] (a simple image descriptor is introduced here referred to as the "image signature". The authors show that it performs better than Itti and Koch model) and GBVS (regularized spatial saliency model of Harel [59]). We also benchmarked our model against spatio-temporal models for saliency prediction in videos like the model proposed by Seo [60] which is built upon optical flow and the model of Wang [9] which is based on the gradient flow field and energy optimization. Finally, the last benchmark is with the ELD [16] deep saliency model which used both high level and low level features for saliency detection under an unified deep learning framework.

In tables 12, 13, 11 below, the comparison of Deep CNN prediction of pixel-wise saliency maps with the Gaze Fixations Density Maps (Gaze-fix) is shown.

The quality of predicted maps is compared with prediction by classical saliency models (Signature Sal, GBVS, Seo) also compared to the same reference: GFDM and the recent state-of-the-art methods of Wang and ELD. The

comparison is given in terms of the widely used AUC metric [61]. Mean value of the metric for each saliency model compared to the GFDM is given together with standard deviation for a sample of videos. In table 11, the $ChaboNet4k$ is compared with reference models for CRCNS data set. In table 12 the maps built on HOLLYWOOD database with its best patch saliency prediction model $Chabonet4K$ are compared with three reference models and in table 13 the comparison is fulfilled on IRCCyN-MVT data set for both 3K and 4K ChaboNet models.

Table 11: The comparison of AUC metric of gaze fixations 'Gaze-fix' vs predicted saliency 'GBVS', 'IttiKoch' , 'Seo' ,'Wang' and 'ELD') and the ChaboNet4k for 890 frames of CRCNS videos.

VideoName	$TotFrame = 890$	Gaze-fix vs GBVS	Gaze-fix vs IttiKoch	Gaze-fix vs Seo	Gaze-fix vs Wang	Gaze-fix vs ELD	Gaze-fix vs ChaboNet4k
beverly03	80	0.78 ± 0.151	0.77 ± 0.124	0.66 ± 0.172	0.73 ± 0.155	0.59 ± 0.118	$\underline{0.79 \pm 0.118}$
gamecube02	303	0.73 ± 0.165	0.74 ± 0.180	0.61 ± 0.179	0.76 ± 0.126	0.78 ± 0.145	$\underline{0.82 \pm 0.126}$
monica05	102	0.75 ± 0.183	0.73 ± 0.158	0.54 ± 0.156	0.73 ± 0.162	0.67 ± 0.162	$\underline{0.79 \pm 0.133}$
standard02	86	$\underline{0.78 \pm 0.132}$	0.72 ± 0.141	0.61 ± 0.169	0.78 ± 0.094	0.60 ± 0.159	0.71 ± 0.181
tv-announce01	73	0.60 ± 0.217	$\underline{0.64 \pm 0.203}$	0.52 ± 0.206	0.00 ± 0.00	0.64 ± 0.209	0.63 ± 0.215
tv-news04	82	0.78 ± 0.169	0.79 ± 0.154	0.61 ± 0.162	0.00 ± 0.00	$\underline{0.85 \pm 0.091}$	0.72 ± 0.145
tv-sports04	164	0.68 ± 0.182	0.69 ± 0.162	0.56 ± 0.193	0.00 ± 0.00	0.64 ± 0.179	$\underline{0.78 \pm 0.172}$

The best AUC metric values are underscored. It can be stated that in general spatial models (Signature Sal, GBVS or Itti) performed better in half of the tested videos. This is due to the fact that these videos contain very contrasted areas in the video frames, which attract human gaze. They do not contain areas having an interesting residual motion. Nevertheless, the *ChaboNet4K* model systematically outperforms Seo's and Wang's model which use motion features. Our proposed deep network still remains competitive with the ELD deep saliency method. This shows definitively that the use of a Deep CNN is a way for prediction of visual saliency in video scenes. However, for IRCCyN-MVT data set, see table 13, despite videos without any motion were left aside, the gain in performance of the proposed model is not very clear due to the complexity of these visual scenes, such as presence of strong contrasts and faces. Using high level features in ELD method ensures an interesting results on IRCCyN-MVT data set.

Table 12: The comparison of AUC metric of gaze fixations 'Gaze-fix' vs predicted saliency 'GBVS', 'SignatureSal' , 'Seo' ,'Wang' and 'ELD') and the ChaboNet4k for the videos from HOLLYWOOD.

VideoName	$TotFrame = 2248$	Gaze-fix vs GBVS	Gaze-fix vs SignatureSal	Gaze-fix vs Seo	Gaze-fix vs Wang	Gaze-fix vs ELD	Gaze-fix vs ChaboNet4k
clipTest56	137	0.76 ± 0.115	0.75 ± 0.086	0.64 ± 0.116	0.68 ± 0.105	0.66 ± 0.105	$\underline{0.77 \pm 0.118}$
clipTest105	154	0.63 ± 0.169	0.57 ± 0.139	0.54 ± 0.123	0.77 ± 0.069	$\underline{0.77 \pm 0.103}$	0.69 ± 0.186
clipTest147	154	0.86 ± 0.093	$\underline{0.90 \pm 0.065}$	0.70 ± 0.103	0.84 ± 0.063	0.76 ± 0.063	0.81 ± 0.146
clipTest250	160	0.74 ± 0.099	0.69 ± 0.110	0.47 ± 0.101	0.81 ± 0.073	$\underline{0.75 \pm 0.116}$	0.71 ± 0.180
clipTest350	66	0.65 ± 0.166	0.68 ± 0.249	0.57 ± 0.124	0.74 ± 0.177	$\underline{0.82 \pm 0.083}$	0.72 ± 0.177
clipTest400	200	0.75 ± 0.127	0.67 ± 0.110	0.60 ± 0.106	0.74 ± 0.133	$\underline{0.74 \pm 0.119}$	0.71 ± 0.146
clipTest451	132	$\underline{0.70 \pm 0.104}$	0.59 ± 0.074	0.57 ± 0.068	0.67 ± 0.096	0.68 ± 0.070	0.63 ± 0.151
clipTest500	166	0.82 ± 0.138	0.84 ± 0.150	0.75 ± 0.152	0.88 ± 0.076	$\underline{0.89 \pm 0.052}$	0.84 ± 0.156
clipTest600	200	0.75 ± 0.131	0.678 ± 0.149	0.53 ± 0.108	$\underline{0.81 \pm 0.091}$	0.72 ± 0.136	0.71 ± 0.180
clipTest650	201	$0.72\pm, 106$	$\underline{0.74 \pm 0.087}$	0.61 ± 0.092	0.65 ± 0.086	0.58 ± 0.090	0.70 ± 0.078
ClipTest700	262	0.74 ± 0.128	0.76 ± 0.099	0.50 ± 0.059	$\underline{0.87 \pm 0.058}$	0.70 ± 0.107	0.78 ± 0.092
clipTest800	200	0.70 ± 0.096	$\underline{0.75 \pm 0.071}$	0.53 ± 0.097	0.56 ± 0.118	0.71 ± 0.128	0.66 ± 0.141
ClipTest803	102	0.86 ± 0.106	0.87 ± 0.068	0.73 ± 0.148	0.85 ± 0.070	0.77 ± 0.088	$\underline{0.88 \pm 0.078}$
ClipTest849	114	0.75 ± 0.155	0.91 ± 0.070	0.55 ± 0.122	$\underline{0.92 \pm 0.024}$	0.86 ± 0.033	0.74 ± 0.132

Table 13: The comparison of AUC metric of gaze fixations 'Gaze-fix' vs predicted saliency 'GBVS', 'SignatureSal' , 'Seo' ,'Wang' and 'ELD') and the ChaboNet3k and ChaboNet4k for the videos from IRCCyN-MVT.

VideoName	$TotFrame = 1227$	Gaze-fix vs GBVS	Gaze-fix vs SignatureSal	Gaze-fix vs Seo	Gaze-fix vs Wang	Gaze-fix vs ELD	Gaze-fix vs ChaboNet3k	Gaze-fix vs ChaboNet4k
src02	37	$0,68 \pm 0,076$	$0,49 \pm 0,083$	$0,44 \pm 0,017$	$0,43 \pm 0,039$	$0,60 \pm 0,043$	$0,012 \pm 0,077$	$0,48 \pm 0,073$
src03	28	$0,82 \pm 0,088$	$0,87 \pm 0,057$	$0,76 \pm 0,091$	$0,80 \pm 0,063$	$0,69 \pm 0,099$	0.00 ± 0.000	$0,70 \pm 0,149$
src04	35	$0,79 \pm 0,058$	$0,81 \pm 0,029$	$0,59 \pm 0,057$	$0,76 \pm 0,112$	$0,81 \pm 0,048$	$0,12 \pm 0,214$	$0,57 \pm 0,135$
src05	35	$0,73 \pm 0,101$	$0,67 \pm 0,122$	$0,48 \pm 0,071$	$0,71 \pm 0,088$	$0,75 \pm 0,113$	$0,39 \pm 0,186$	$0,53 \pm 0,128$
src06	36	$0,85 \pm 0,080$	$0,71 \pm 0,151$	$0,73 \pm 0,148$	$0,81 \pm 0,092$	$0,72 \pm 0,153$	0.00 ± 0.000	$0,60 \pm 0,180$
src07	36	$0,72 \pm 0,070$	$0,73 \pm 0,060$	$0,57 \pm 0,060$	$0,74 \pm 0,062$	$0,65 \pm 0,090$	$0,34 \pm 0,284$	$0,55 \pm 0,135$
src10	33	$0,87 \pm 0,048$	$0,92 \pm 0,043$	$0,82 \pm 0,101$	$0,90 \pm 0,045$	$0,82 \pm 0,074$	$0,90 \pm 0,045$	$0,60 \pm 0,173$
src13	35	$0,79 \pm 0,103$	$0,75 \pm 0,111$	$0,64 \pm 0,144$	$0,82 \pm 0,067$	$0,71 \pm 0,116$	$0,36 \pm 0,201$	$0,52 \pm 0,138$
src17	42	$0,55 \pm 0,092$	$0,33 \pm 0,099$	$0,45 \pm 0,033$	$0,45 \pm 0,062$	$0,62 \pm 0,095$	0.00 ± 0.000	$0,51 \pm 0,098$
src19	33	$0,76 \pm 0,094$	$0,68 \pm 0,086$	$0,59 \pm 0,117$	$0,75 \pm 0,062$	$0,89 \pm 0,020$	$0,46 \pm 0,075$	$0,75 \pm 0,123$
src23	40	$0,76 \pm 0,050$	$0,69 \pm 0,070$	$0,58 \pm 0,067$	$0,77 \pm 0,045$	$0,83 \pm 0,027$	$0,03 \pm 0,169$	$0,66 \pm 0,105$
src24	33	$0,63 \pm 0,071$	$0,58 \pm 0,054$	$0,55 \pm 0,059$	$0,65 \pm 0,040$	$0,55 \pm 0,046$	$0,23 \pm 0,252$	$0,50 \pm 0,052$
src27	33	$0,59 \pm 0,117$	$0,64 \pm 0,091$	$0,52 \pm 0,057$	$0,77 \pm 0,064$	$0,68 \pm 0,071$	0.00 ± 0.000	$0,54 \pm 0,106$

Table 14: Visual evaluation of different saliency models for video taken from Hollywood data set '$NameVideo\#frameNumber$'

Table 15, presents the time needed for testing one patch and the creation of the saliency map across one frame with a stride of 5 pixels. We used machines with Xeon E5 processor. The computation time for one patch is quite reasonable, the overall time for one full HD video frame remains high despite parallelization of computations. This is explained by a large quantity of scanning windows, more GPU processors are need for faster computation.

Table 15: Time for testing one patch and one frame of video.

	machine $8\mu p$	machine $20\mu p$	machine $2 \times 14 cores \mu p$
patch 100×100	0.015s	0.028s	0.011s
frame 720×576	42.31s	18.49s	8.56s

7. Conclusion and perspectives

This study addressed the problem of prediction of visual attention in video content with Deep CNNs. In this paper we have further extended experiments and confirmed partial results we obtained in our previous works. We hypothesized that the model could capture gaze attraction by moving objects due to the residual motion maps added to primary color pixel values. First of all, we measured the correspondence of residual motion maps with gaze fixations of observes which confirmed the interest of their incorporation into input layer of proposed Deep architecture. The performances of prediction when different kinds of features are ingested by the network -color pixel values only, color values with residual motion- were compared. As far as dynamic content is concerned, the saliency is better predicted with spatio-temporal features (RGB and residual motion) when scenes do not contain distracting contrasts. The proposed relatively shallow architecture ChaboNet was compared to similar architectures AlexNet and LeNet and showed better prediction power in terms of mean accuracy and stability of training phase. The transfer learning scheme applied to the prediction of saliency on small data sets by fine-tuning parameters pre-trained on

a large data set (HOLLYWOOD) successfully outperforms the state-of-the-art, i.e. Bengio's method. Finally, a method for building pixel-wise saliency maps, using the probability of patches to be salient, was extensively tested against reference spatial, spatio-temporal and a deppe learning-based prediction models. We come to the conclusion that the interpolation of classification results is not a definite way to build dense predicted saliency maps. Indeed, accuracy in patch classification are high, but the AUC metric values with reference GFDMs are not systematically better than reference classical models. We hypothesize that this is partly a distractor problem, but still the way to predict a dense map can be further developed with FC-networks. Last but not least, Deep NNs supply an interesting framework for use of temporal continuity of saliency maps in the absence of distractors. This is the future of the present research.

Acknowledgment

This research has been supported by University of Bordeaux, University of Sfax and the grant UNetBA.

References

[1] A. M. Treisman, G. Gelade, A feature-integration theory of attention, Cognitive Psychology 12 (1) (1980) 97–136.

[2] B. J. Han, J. Y. Sim, Saliency detection for panoramic landscape images of outdoor scenes, Journal of Visual Communication and Image Representation 49 (Supplement C) (2017) 27 – 37.

[3] Z. Li, C. Lang, S. Feng, T. Wang, Saliency ranker: A new salient object detection method, Journal of Visual Communication and Image Representation 50 (Supplement C) (2018) 16 – 26.

[4] J. Ren, Z. Liu, X. Zhou, G. Sun, C. Bai, Saliency integration driven by similar images, Journal of Visual Communication and Image Representation 50 (2018) 227 – 236.

[5] Z. Liu, X. Zhang, S. Luo, O. Le Meur, Superpixel-based spatiotemporal saliency detection, IEEE Transactions on Circuits and Systems for Video Technology 24 (9) (2014) 1522–1540. doi:10.1109/TCSVT.2014.2308642.

[6] J. Yang, G. Zhao, J. Yuan, X. Shen, Z. Lin, B. L. Price, J. Brandt, Discovering primary objects in videos by saliency fusion and iterative appearance estimation, IEEE Trans. Circuits Syst. Video Techn. 26 (6) (2016) 1070–1083. doi:10.1109/TCSVT.2015.2433171.
URL http://dx.doi.org/10.1109/TCSVT.2015.2433171

[7] Y. Pang, L. Ye, X. Li, J. Pan, Incremental learning with saliency map for moving object detection, IEEE Transactions on Circuits and Systems for Video Technology PP (99) (2016) 1–1. doi:10.1109/TCSVT.2016.2630731.

[8] L. Duan, T. Xi, S. Cui, H. Qi, A. C. Bovik, A spatiotemporal weighted dissimilarity-based method for video saliency detection, Signal Processing: Image Communication 38 (2015) 45 – 56, recent Advances in Saliency Models, Applications and Evaluations. doi:http://dx.doi.org/10.1016/j.image.2015.08.005.
URL http://www.sciencedirect.com/science/article/pii/S0923596515001307

[9] W. Wang, J. Shen, L. Shao, Consistent video saliency using local gradient flow optimization and global refinement, IEEE Transactions on Image Processing 24 (11) (2015) 4185–4196.

[10] S. Zhong, Y. Liu, F. Ren, J. Zhang, T. Ren, Video saliency detection via dynamic consistent spatio-temporal attention modelling, in: Proceedings of the Twenty-Seventh AAAI Conference on Artificial Intelligence, AAAI Press, 2013, pp. 1063–1069.

[11] I. González-Díaz, V. Buso, J. Benois-Pineau, Perceptual modeling in the problem of active object recognition in visual scenes, Pattern Recognition 56 (2016) 129–141.

[12] Y. Pinto, A. R. van der Leij, I. G. Sligte, V. F. Lamme, H. S. Scholte, Bottom-up and top-down attention are independent, Journal of Vision 13 (3) (2013) 16.

[13] J. Shen, L. Itti, Top-down influences on visual attention during listening are modulated by observer sex, Vision Research 65 (2012) 62–76.

[14] A. Borji, D. N. Sihite, L. Itti, What stands out in a scene? A study of human explicit saliency judgment, Vision Research 91 (2013) 62–77.

[15] H. Boujut, J. Benois-Pineau, T. Ahmed, O. Hadar, P. Bonnet, No-reference video quality assessment of h.264 video streams based on semantic saliency maps, Vol. 8293, 2012, pp. 82930T–82930T–9.

[16] G. Lee, Y.-W. Tai, J. Kim, Deep saliency with encoded low level distance map and high level features, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 660–668.

[17] G. Sharma, F. Jurie, C. Schmid, Discriminative spatial saliency for image classification, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3506–3513.

[18] L. Deng, D. Yu, DEEP LEARNING: Methods and Applications, Foundations and Trends in Signal Processing 7 (MSR-TR-2014-21) (2014) 197–387.

[19] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, Gradient-Based Learning Applied to Document Recognition, no. 86(11), 1998, pp. 2278–2324.

[20] J. Bruna, S. Mallat, Invariant Scattering Convolution Networks, IEEE Trans. Pattern Anal. Mach. Intell. 35 (8) (2013) 1872–1886.

[21] Y. Bengio, A. Courville, P. Vincent, Representation Learning: A Review and New Perspectives, IEEE Transactions on Pattern Analysis and Machine Intelligence (35 (8)) (2014) 1798–1828.

[22] A. Krizhevsky, I. Sutskever, G. E. Hinton, ImageNet Classification with Deep Convolutional Neural Networks, in: F. Pereira, C. Burges, L. Bottou,

K. Weinberger (Eds.), Advances in Neural Information Processing Systems 25, Curran Associates, Inc., 2012, pp. 1097–1105.

[23] H. R. Tavakoli, A. Borji, J. Laaksonen, E. Rahtu, Exploiting inter-image similarity and ensemble of extreme learners for fixation prediction using deep features, CoRR abs/1610.06449.
URL http://arxiv.org/abs/1610.06449

[24] S. F. Dodge, L. J. Karam, Visual saliency prediction using a mixture of deep neural networks, CoRR abs/1702.00372. arXiv:1702.00372.

[25] R. Monroy, S. Lutz, T. Chalasani, A. Smolic, Salnet360: Saliency maps for omni-directional images with CNN, CoRR abs/1709.06505. arXiv:1709.06505.
URL http://arxiv.org/abs/1709.06505

[26] S. S. S. Kruthiventi, V. Gudisa, J. H. Dholakiya, R. V. Babu, Saliency Unified: A Deep Architecture for simultaneous Eye Fixation Prediction and Salient Object Segmentation, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) 00 (2016) 5781–5790. doi:10.1109/CVPR.2016.623.

[27] M. Kümmerer, L. Theis, M. Bethge, Deep Gaze I: Boosting Saliency Prediction with Feature Maps Trained on ImageNet, CoRR abs/1411.1045.

[28] J. Pan, Giró i Nieto, X., End-to-end Convolutional Network for Saliency Prediction, CoRR abs/1507.01422.

[29] C. Shen, Q. Zhao, Learning to Predict Eye Fixations for Semantic Contents Using Multi-layer Sparse Network, Neurocomputing 138 (2014) 61–68.

[30] K. Simonyan, A. Vedaldi, A. Zisserman, Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps, CoRR abs/1312.6034.

[31] E. Vig, M. Dorr, D. Cox, Large-Scale Optimization of Hierarchical Features for Saliency Prediction in Natural Images , in: Proceedings of the 2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR '14, 2014, pp. 2798–2805.

[32] J. Han, L. Sun, X. Hu, J. Han, L. Shao, Spatial and temporal visual attention prediction in videos using eye movement data, Neurocomputing 145 (2014) 140–153.

[33] S. Chaabouni, J. Benois-Pineau, C. Ben Amar, Transfer learning with deep networks for saliency prediction in natural video, in: 2016 IEEE International Conference on Image Processing (ICIP), Vol. 91, 2016, pp. 1604–1608.

[34] A. Coutrot, O. Le Meur, Visual attention saccadic models: taking into account global scene context and temporal aspects of gaze behaviour, 2016, poster.
URL https://hal.inria.fr/hal-01391751

[35] M. Cornia, L. Baraldi, G. Serra, R. Cucchiara, Multi-level net: A visual saliency prediction model, in: Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part II, 2016, pp. 302–315. doi:10.1007/978-3-319-48881-3_21.
URL https://doi.org/10.1007/978-3-319-48881-3_21

[36] D. S. Wooding, Eye movements of large populations: II. Deriving regions of interest, coverage, and similarity using fixation maps, Behavior Research Methods, Instruments, & Computers 34 (4) (2002) 518–528.

[37] F. Boulos, W. Chen, B. Parrein, P. Le Callet, Region-of-Interest Intra Prediction for H.264/AVC Error Resilience, in: IEEE International Conference on Image Processing, Cairo, Egypt, 2009, pp. 3109–3112.

[38] S. Chaabouni, J. Benois-Pineau, O. Hadar, Prediction of visual saliency in video with deep cnns, Vol. 9971, 2016, pp. 99711Q–99711Q–14.

[39] L. Mai, H. Le, Y. Niu, F. Liu, Rule of thirds detection from photograph, in: Multimedia (ISM), 2011 IEEE International Symposium on, 2011, pp. 91–96.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going Deeper with Convolutions, in: Computer Vision and Pattern Recognition (CVPR), 2015.

[41] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition, CoRR abs/1409.1556. arXiv:1409.1556.
URL http://arxiv.org/abs/1409.1556

[42] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016) 770–778.

[43] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: Convolutional Architecture for Fast Feature Embedding, in: Proceedings of the ACM International Conference on Multimedia, MM '14, Orlando, FL, USA, November 03 - 07, 2014, 2014, pp. 675–678.

[44] A. Borji, L. Itti, State-of-the-art in Visual Attention Modeling, IEEE Transactions on Pattern Analysis and Machine Intelligence 35 (1) (2013) 185–207.

[45] S. Marat, T. Phuoc, L. Granjon, N. Guyader, D. Pellerin, A. Guerin-Dugue, Modelling spatiotemporal saliency to predict gaze direction for short videos., International Journal of Computer Vision (82) (2009) 231–243.

[46] D. Purves, G. J. Augustine, D. Fitzpatrick, L. C. Katz, A. S. LaMantia, J. O. McNamara, S. M. Williams, Neuroscience, 2nd edition, Sinauer Associates, 2001.

[47] M. A. Fischler, R. C. Bolles, Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography, Magazine Communications of the ACM 24 (6) (1981) 381–395.

[48] A. Krizhevsky, Learning multiple layers of features from tiny images, Ph.D. thesis, University of Toronto (2009).

[49] S. Chaabouni, J. Benois-Pineau, F. Tison, C. Ben Amar, Prediction of visual attention with deep CNN for studies of neurodegenerative diseases, in: 2016 14th International Workshop on Content-Based Multimedia Indexing (CBMI), 2016, pp. 1–6.

[50] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks?, in: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, K. Weinberger (Eds.), Advances in Neural Information Processing Systems 27, Curran Associates, Inc., 2014, pp. 3320–3328.

[51] M. D. Zeiler, R. Fergus, Visualizing and Understanding Convolutional Networks, in: D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars (Eds.), Computer Vision ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I, Springer International Publishing, Cham, 2014, pp. 818–833.

[52] G. Mesnil, Y. Dauphin, X. Glorot, S. Rifai, Y. Bengio, I. J. Goodfellow, E. Lavoie, X. Muller, G. Desjardins, D. Warde-Farley, P. Vincent, A. Courville, J. Bergstra, Unsupervised and Transfer Learning Challenge: a Deep Learning approach, in: JMLR W& CP: Proceedings of the Unsupervised and Transfer Learning challenge and workshop, Vol. 27, 2012, pp. 97–110.

[53] M. Marszałek, I. Laptev, C. Schmid, Actions in Context, in: 2009 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2929–2936.

[54] S. Mathe, C. Sminchisescu, Actions in the Eye: Dynamic Gaze Datasets

and Learnt Saliency Models for Visual Recognition, IEEE Transactions on Pattern Analysis and Machine Intelligence 37 (7) (2015) 1408–1424.

[55] L. Itti, CRCNS Data Sharing: Eye movements during free-viewing of natural videos, in: Collaborative Research in Computational Neuroscience Annual Meeting, Los Angeles, California, 2008.

[56] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, 2015, pp. 448–456.

[57] L. Itti, C. Koch, E. Niebur, A Model of Saliency-Based Visual Attention for Rapid Scene Analysis, IEEE Transactions on Pattern Analysis and Machine Intelligence 20 (11) (1998) 1254–1259.

[58] X. Hou, J. Harel, C. Koch, Image Signature: Highlighting Sparse Salient Regions., IEEE Trans. Pattern Anal. Mach. Intell. 34 (1) (2012) 194–201.

[59] J. Harel, C. Koch, P. Perona, Graph-based visual saliency, in: Advances in neural information processing systems 19, MIT Press, 2007, pp. 545–552.

[60] H. J. Seo, P. Milanfar, Static and space-time visual saliency detection by self-resemblance, Journal of Vision (9(12):15) (2009) 1–27.

[61] O. Le Meur, T. Baccino, Methods for comparing scanpaths and saliency maps: strengths and weaknesses, Behavior Research Methods 45 (1) 251–266.