

# How to Make an Image More Memorable? A Deep Style Transfer Approach

Aliaksandr Siarohin<sup>1</sup>, Gloria Zen<sup>1</sup>, Cveta Majtanovic<sup>2</sup>,  
Xavier Alameda-Pineda<sup>3</sup>, Elisa Ricci<sup>4</sup> and Nicu Sebe<sup>1</sup>

<sup>1</sup>University of Trento, name.lastname@unitn.it

<sup>2</sup>University of Novi Sad and University of Trento, cveta.majtanovic@unitn.it

<sup>3</sup>INRIA Grenoble, xavier.alameda-pineda@inria.fr

<sup>4</sup>Fondazione Bruno Kessler and University of Perugia, eliricci@fbk.eu

## ABSTRACT

Recent works have shown that it is possible to automatically predict intrinsic image properties like memorability. In this paper, we take a step forward addressing the question: “Can we make an image more memorable?”. Methods for automatically increasing image memorability would have an impact in many application fields like education, gaming or advertising. Our work is inspired by the popular *editing-by-applying-filters* paradigm adopted in photo editing applications, like Instagram and Prisma. In this context, the problem of increasing image memorability maps to that of retrieving “memorabilizing” filters or style “seeds”. Still, users generally have to go through most of the available filters before finding the desired solution, thus turning the editing process into a resource and time consuming task. In this work, we show that it is possible to automatically retrieve the *best* style seeds for a given image, thus remarkably reducing the number of human attempts needed to find a good match. Our approach leverages from recent advances in the field of image synthesis and adopts a deep architecture for generating a memorable picture from a given input image and a style seed. Importantly, to automatically select the best style a novel learning-based solution, also relying on deep models, is proposed. Our experimental evaluation, conducted on publicly available benchmarks, demonstrates the effectiveness of the proposed approach for generating memorable images through automatic style seed selection.

## CCS Concepts

•Computing methodologies → Computer vision; Computational photography;

## Keywords

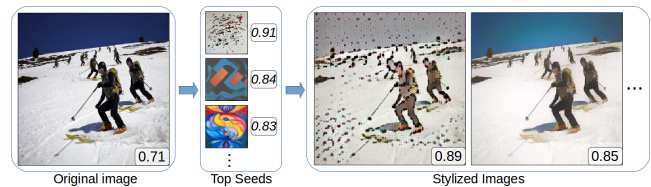
Memorability; photo enhancement; deep style transfer; retrieval.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICMR '17, June 06-09, 2017, Bucharest, Romania

© 2017 ACM. ISBN 978-1-4503-4701-3/17/06...\$15.00

DOI: <http://dx.doi.org/10.1145/3078971.3078986>



**Figure 1:** Sample results illustrating our idea (best viewed in colors). Given a generic image (left), automatically find the best style filters (center) that augment the memorability of the image (right). Memorability scores in the range [0,1] are reported in the small boxes.

## 1. INTRODUCTION

Today’s expansion of infographics is certainly related to one of the everyday life idiom “A picture is worth a thousand words” (or more precisely 84.1 words [4]) and to the need of providing the fastest possible knowledge transfer in the current “information overload” age. A recent study [13] showed that everyone is bombarded by the equivalent of 174 newspapers of data every day. In this context, we ask ourselves: *Is it possible to transform a user-chosen image so that it has more chances to be remembered?*

For this question to be properly stated, it requires the existence of a measure of memorability, and recent studies proved that memorability is intrinsic to the visual content and is measurable [16, 17]. Indeed, these studies use *memory games* to provide an objective evaluation of image memorability, which has surprisingly low variance across trials. Recent studies have also provided tools to detect the visual features responsible for both memorable and easily forgettable images. For instance, images that tend to be forgotten lack distinctiveness, like natural landscapes, whereas pictures with people, specific actions and events or central objects are way more memorable [6]. Previous papers have also analyzed the relationship between emotions and memorability [8]. In a similar line of thought, researchers wondered how to accurately predict which images will be remembered and which will be not. Recent experiments showed near-human performance in estimating, measuring and predicting visual memorability [19], where MemNet, a model trained on the largest annotated image memorability dataset, LaMem, has been proposed.

While previous studies on automatic prediction of memorability from images paved the way towards the automatic recognition of image memorability, many questions are still open. For instance: *is it possible to increase the memorabil-*

ity of an image, while keeping its high-level content? Imagine an advertising campaign concerning the design for a new product targeting a specific market sector. Once the very expensive designing phase is over, the client receives a set of images advertising the new product. Such images tell a story: in the attempt of increasing the image’s memorability, the high-level content, that is *the meaning*, should remain intact. We therefore focus on how to automatically modify the style of the image, that is how to *filter* the image, so as to make it more memorable.

Some popular commercial products are based on this image filtering philosophy, for other purposes than memorability though. For instance, Instagram<sup>1</sup>, a photo and video sharing Internet platform launched in 2010, allows the users to filter the visual content with several pre-defined filters before sharing. Similarly, Prisma<sup>2</sup> turns user memories into art by using artificial intelligence. In parallel to the development of these commercial products, several recent research studies in computer vision and multimedia have focused on creating artistic images of high perceptual quality with artificial intelligence models. For instance, Gatys *et al.* [10] have proposed an approach where a deep network is used to manipulate the content of a natural image adapting it to the style of a given artwork. Subsequently, more efficient deep architectures for implementing a style transfer have been introduced [30]. Importantly, none of these commercial products and related research studies incorporate the notion of image memorability.

In this work, we propose a novel approach for increasing the memorability of images which is inspired by the *editing-by-filtering* framework (Fig. 1). Our method relies on three deep networks. A first deep architecture, the *Synthesizer* network, is used to synthesize a memorable image from the input picture and a style picture. A second network acts as a style *Selector* and it is used to retrieve the “best” style seed to provide to the Synthesizer, (*i.e.* the one that will produce the highest increase in terms of memorability) given the input picture. To train the Selector, pairs of images and vectors of memorability gap scores (indicating the increase/decrease in memorability when applying each seed to the image) are used. A third network, the *Scorer*, which predicts the memorability score from a given input image, is used to compute the memorability gaps necessary to train the Selector. Our approach is extensively evaluated on the publicly available LaMem dataset [19] and we show that it can be successfully used to automatically increase the memorability of natural images.

The main contributions of our work are the following:

- We tackle the challenging task of increasing image memorability while keeping the high-level content intact (thus modifying only the style of the image).
- We cast this into a style-based image synthesis problem using deep architectures and propose an automatic method to retrieve the style seeds that are expected to lead to the largest increase of memorability for the input image.
- We propose a lightweight solution for training the Selector network implementing the style seed selection process, allowing us to efficiently learn our model with a reduced number of training data while considering relatively large variations of style pictures.

<sup>1</sup><https://www.instagram.com>

<sup>2</sup><http://prisma-ai.com>

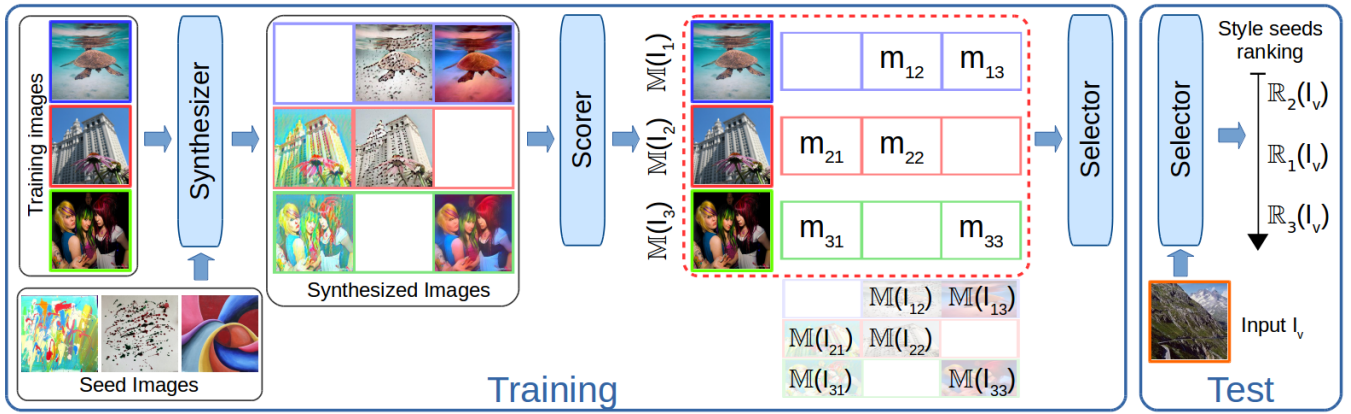
## 2. RELATED WORKS

The concept of memorability and its relation with other aspects of the human mind has been long studied from a psychological perspective [25, 3, 14, 28]. Works in psychology and neuroscience mostly focused on visual memory, studying for instance the human capacity of remembering object details [6], the effect of emotions on memory [3, 28] or the brain’s learning mechanisms, *e.g.* the role of the amygdala in memory [25, 28]. For a few years now, more automated studies on memorability have arisen: from the collection of image datasets specifically designed to study memorability, to user-friendly techniques to annotate these data with memorability scores. The community is now paying attention to understand the causes of visual memorability and its prominent links with, for instance, image content, low- and mid-level visual features and evoked emotions.

Isola *et al.* [16] showed that visual memorability is an intrinsic property of images, and that it can be explained by considering only image visual features. Besides the expected inter-subject variability, [16] reported a large consistency among viewers when measuring the memorability of several images. Typically, such measures are obtained by playing a memory game. Other studies proved that memorability can also be automatically predicted. Recently, Khosla *et al.* [19] used CNN-based features from MemNet to achieve a prediction accuracy very close to human performance, *i.e.* up to the limit of the inter-subject variability, thus outperforming previous works using hand-crafted features such as objects or visual attributes [17].

In parallel, large research efforts have been invested in understanding what makes an image memorable and, in a complementary manner, which is the relation between image memorability and other subjective properties of visual data, such as interestingness, aesthetics or evoked emotions. Gygli *et al.* [12] observed that memorability negatively correlates with visual interestingness. Curiously, they also showed that human beings perform quite bad at judging the memorability of an image, thus further justifying the use of memory games for annotation. In the same study, it was shown that aesthetic, visual interestingness and human judgements of memorability are highly correlated. Similar results were reported later on in [17], confirming these findings. A possible mundane interpretation of these findings is that people wish to remember what they like or find interesting, though this is not always the case.

Khosla *et al.* [19] showed that, with the exception of *amusement*, images that evoke negative emotions like *disgust*, *anger* and *fear* are more likely to be remembered. Conversely, images that evoke emotions like *awe* and *contentment* tend to be less memorable. Similarly, the authors of [15] showed that attributes like *peaceful* are negatively correlated with memorability. Other works showed that *arousal* has a strong effect on human memory [3, 8, 26] at two different stages: either during the encoding of visual information (*e.g.*, increased attention and/or processing) or post-encoding (*e.g.*, enhanced consolidation when recalling the stored visual information). Memorability was also investigated with respect to distinctiveness and low-level cues such as colors in [5] and with respect to eye fixation in [19, 7]. In more detail, [5] discussed how images that stand out of the context (*i.e.*, they are unexpected or unique) are more easily remembered and that memorability significantly depends upon the number of distinct colors in the image. These findings support



**Figure 2: Overview of our method.** At training time, the Synthesizer  $\mathcal{S}$  and the Scorer  $\mathcal{M}$  serve to generate the training data (highlighted with a red dotted frame) for the seed Selector  $\mathcal{R}$ . At test time, the seed Selector provides for each new image a sorted list of style seeds, based on the predicted memorability increase  $\mathcal{R}_s(I_v)$ .

our intuition that it is possible to manipulate an image to increase its memorability. Indeed, this can happen for example by indirectly modifying image distinctiveness or the evoked arousal. Along the same line of thought, Peng *et al.* [27] attempted to modify the emotions evoked by an image adjusting its color tone and its texture-related features.

Recent works analyzed how images can be modified to increase or decrease their memorability [18, 19]. These are based on other contemporary studies that focused on generating memorability maps of images [21, 22, 20]. In particular, Khosla *et al.* [19] showed that by removing visual details from an image through a cartoonization process the memorability score can be modified. However, they did not provide a methodology to systematically increase the memorability of pictures. The same group [18] also demonstrated that it is possible to increase the memorability of faces, while maintaining the identity of the person and properties like age, attractiveness and emotional magnitude. Up to our knowledge, this is the first attempt to automatically increase the memorability of generic images (not only faces).

### 3. METHOD

In this section we introduce the proposed framework to automatically increase the memorability of an input image. Our method is designed in a way such that the process of “memorabilizing” images is performed in an efficient manner while preserving most of the high-level image content.

#### 3.1 Overview

The proposed approach co-articulates three main components, namely: the seed Selector, the Scorer and the Synthesizer, and so we refer to it as  $S^3$  or *S-cube*. In order to give a general idea of the overall methodological framework, we illustrate the pipeline associated to *S-cube* in Figure 2. The Selector is the core of our approach: for a generic input image  $\mathbf{I}$  and given a set of style image seeds  $\mathcal{S}$ , the Selector retrieves the subset of  $\mathcal{S}$  that will be able to produce the largest increase of memorability. In details, the seed Selector predicts the expected increase/decrease of memorability that each seed  $\mathbf{S} \in \mathcal{S}$  will produce in the input image  $\mathbf{I}$ , and consequently it ranks the seeds according to the expected increase of memorability. At training time, the Synthesizer and the Scorer are used to generate images from many input image-seed pairs and to score these pairs,

respectively. Each input image is then associated to the relative increase/decrease of memorability obtained with each of the seeds. With this information, we can learn to predict the increase/decrease of memorability for a new image, and therefore rank the seeds according to the expected increase. Indeed, at query time, the Selector is able to retrieve the most memorabilizing seeds and give them to the Synthesizer. In the following, we first formalize the *S-cube* framework and then describe each of the three components in detail.

#### 3.2 The S-cube approach

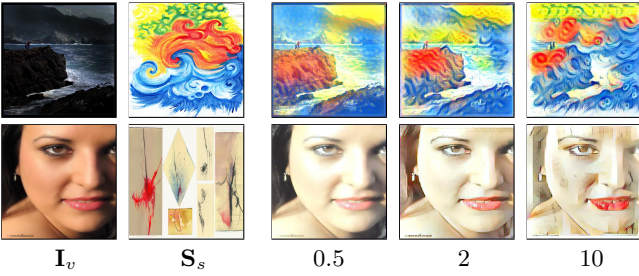
Let us denote the Scorer, the Synthesizer and the seed Selector models by  $\mathcal{M}$ ,  $\mathcal{S}$  and  $\mathcal{R}$ , respectively. During the **training phase** the three models are learned. The Scoring model  $\mathcal{M}$  returns the memorability value  $\mathcal{M}(\mathbf{I})$  of a generic image  $\mathbf{I}$ , and it is learned by means of a training set of images annotated with memorability:  $\mathcal{M} = \{\mathbf{I}_i^{\mathcal{M}}, m_i\}_{i=1}^I$ . In addition to this training set, we also consider a generating set of natural images  $\mathcal{G} = \{\mathbf{I}_g^{\mathcal{G}}\}_{g=1}^G$  and a set of style seed images  $\mathcal{S} = \{\mathbf{S}_s\}_{s=1}^S$ . The Synthesizer produces an image from an image-seed pair,

$$\mathbf{I}_{gs} = \mathcal{S}(\mathbf{I}_g^{\mathcal{G}}, \mathbf{S}_s). \quad (1)$$

The scoring model  $\mathcal{M}$  and the Synthesizer  $\mathcal{S}$  are the required steps to train the seed Selector  $\mathcal{R}$ . Indeed, for each image  $\mathbf{I}_g^{\mathcal{G}} \in \mathcal{G}$  and for each style seed  $\mathbf{S}_s \in \mathcal{S}$ , the synthesis procedure generates  $\mathbf{I}_{gs}$ . The Scoring model is used to compute the memorability score gap between the synthesized and the original images:

$$m_{gs}^{\mathcal{M}} = \mathcal{M}(\mathbf{I}_{gs}) - \mathcal{M}(\mathbf{I}_g^{\mathcal{G}}). \quad (2)$$

The seed-wise concatenation of these scores, denoted by  $\mathbf{m}_g^{\mathcal{M}} = (m_{gs}^{\mathcal{M}})_{s=1}^S$ , is used to learn the seed Selector. Specifically, a training set of natural images labeled with the seed-wise concatenation of memorability gaps  $\mathcal{R} = \{\mathbf{I}_g^{\mathcal{G}}, \mathbf{m}_g^{\mathcal{M}}\}_{g=1}^G$  is constructed. The process of seed selection is casted as a regression problem and the mapping  $\mathcal{R}$  between an image and the associated vector of memorability gap scores is learned. This indirectly produces a ranking of the seeds in terms of their ability to memorabilize images (*i.e.* the best seed corresponds to the largest memorability increase).



**Figure 3: Sample results.** (Left) Original images and applied style seeds. (Right) Synthesized images at varying parameter  $\alpha$ , which regulates the trade-off between preserving the original content of the given image  $\mathbf{I}_v$  and transferring the style  $\mathbf{S}_s$ .

During the **test phase** and given a novel image  $\mathbf{I}_v$ , the seed Selector is applied to predict the vector of memorability gap scores associated to all style seeds, *i.e.*  $\mathbf{m}_v = \mathbb{R}(\mathbf{I}_v)$ . A ranking of seeds is then derived from the vector  $\mathbf{m}_v$ . Based on this ranking the Synthesizer is applied to the test image  $\mathbf{I}_v$  considering only the top  $Q$  style seeds  $\mathbf{S}_s$  and produces a set of stylized images  $\{\mathbf{I}_{qs}\}_{q=1}^Q$ .

In the following we describe the three main building blocks of our approach, providing details of our implementation.

### 3.3 The Scorer

The scoring model  $\mathbb{M}$  returns an estimate of the memorability associated to an input image  $\mathbf{I}$ . In our work, we use the memorability predictor based on LaMem in [19], which is the state of the art to automatically compute image memorability. In details, following [19] we consider a hybrid CNN model [31]. The network is pre-trained first for the object classification task (*i.e.* on ImageNet database) and then for the scene classification task (*i.e.* on Places dataset). Then, we randomly split the LaMem training set into two disjoint subsets (of 22,500 images each),  $\mathcal{M}$  and  $\mathcal{E}$ . We use the pre-trained model and the two subsets to learn two independent scoring models  $\mathbb{M}$  and  $\mathbb{E}$ . While, as discussed above,  $\mathbb{M}$  is used during the training phase of our approach, the model  $\mathbb{E}$  is adopted for evaluation (see Section 4.1). For training, we run 70k iterations of stochastic gradient descent with momentum 0.9, learning rate  $10^{-3}$  and batch size 256.

### 3.4 The Synthesizer

The Synthesizer takes as an input a generic image  $\mathbf{I}_g$  and a style seed image  $\mathbf{S}_s$  and produces a stylized image  $\mathbf{I}_{gs} = \mathbb{S}(\mathbf{I}_g, \mathbf{S}_s)$ . We use the strategy proposed in [30], which consists on training a different feed-forward network for every seed. As seeds, we use 100 abstract paintings from the DeviantArt database [29], and therefore we train  $S = 100$  networks for 10k iterations with learning rate  $10^{-2}$ . The most important hyper-parameter is the coefficient  $\alpha$ , which regulates the trade-off between preserving the original image content and producing something closer to the style seed (see Figure 3). In our experiments we evaluated the effect of  $\alpha$  (see Section 4). It is worth noticing that the methodology proposed in this article is independent of the synthesis procedure. Indeed, we also tried other methods, namely Gatys *et al.* [9] and Li *et al.* [23], but we selected [30] since it showed very good performance while keeping low computational complexity. This is especially important in our

framework since the Synthesizer is also used to generate the training set for learning  $\mathbb{R}$ .

### 3.5 The Seed Selector

The core part of our approach is the Selector. Given a training set of natural images labeled with the vector of memorability gaps:  $\mathcal{R} = \{\mathbf{I}_g^{\mathcal{G}}, \mathbf{m}_g^{\mathbb{M}}\}_{g=1}^G$ , the seed Selector  $\mathbb{R}$  is trained minimizing the following objective:

$$\mathcal{L}_{\mathbb{R}} = \sum_{g=1}^G \mathcal{L}(\mathbb{R}(\mathbf{I}_g^{\mathcal{G}}), \mathbf{m}_g^{\mathbb{M}}). \quad (3)$$

where  $\mathcal{L}$  is a loss function which measures the discrepancy between the learned vector  $\mathbb{R}(\mathbf{I}_g^{\mathcal{G}})$  and the memorability gap scores  $\mathbf{m}_g^{\mathbb{M}}$ . By training the seed Selector with memorability gaps, we are learning *by how much each of the seeds increases or decreases the memorability of a given image*. This has several advantages. First, we can very easily rank the seeds by the expected increase in memorability they will produce if used together with the input image and the synthesis procedure. Second, if several seeds have similar expected memorability increase, they can be proposed to the user for further selection. Third, if all seeds are expected to decrease the memorability, the optimal choice of not modifying the image can easily be made. Fourth, once  $\mathbb{R}$  is trained, all this information comes at the price of evaluating  $\mathbb{R}$  for a new image, which is cheaper than running  $\mathbb{S}$  and  $\mathbb{M}$   $S$  times.

Even if this strategy has many advantages at testing time, the most prominent drawback is that, to create the training set  $\mathcal{R}$ , one should ideally call the synthesis procedure for all possible image-seed pairs. This clearly reduces the scalability and the flexibility of the proposed approach. The scalability because training the model on a large image dataset means generating a much larger dataset (*i.e.*,  $S$  times larger). The flexibility because if one wishes to add a new seed to the set  $\mathcal{S}$ , then all image-seed pairs for the new seed need to be synthesized and this takes time. Therefore, it would be desirable to find a way to overcome these limitations while keeping the advantages described in the previous paragraph.

The solution to these issues comes with a model able to learn from a partially synthesized set, in which not all image-seed pairs are generated and scored. This means that the memorability gap vector  $\mathbf{m}_g^{\mathbb{M}}$  has missing entries. In this way we only require to generate *enough* image-seed pairs. To this aim, we propose to use a decomposable loss function  $\mathcal{L}$ . Formally, we define a binary variable  $\omega_{gs}$  set to 1 if the  $gs$ -th image-seed pair is available and to 0 otherwise and rewrite the objective function in (3) as:

$$\mathcal{L}_{\mathbb{R}} = \sum_{g=1}^G \sum_{s=1}^S \omega_{gs} \ell(\mathbb{R}_s(\mathbf{I}_g^{\mathcal{G}}), m_{gs}^{\mathbb{M}}). \quad (4)$$

where  $\mathbb{R}_s$  is the  $s$ -th component of  $\mathbb{R}$  and  $\ell$  is the square loss. We implemented this model using an AlexNet architecture, where the prediction errors for the missing entries of  $\mathbf{m}_g^{\mathbb{M}}$  are not back-propagated. Specifically, we considered the pre-trained Hybrid-CNN and fine-tune only the layers fc6, fc7, conv5, conv4 using learning rate equal to  $10^{-3}$ , momentum equal to 0.9 and batch size 64. The choice of Hybrid-CNN is considered more appropriate when dealing with generic images since the network is pre-trained both on images of places and objects.

$\alpha$	$\bar{\omega}$	$A^{\mathbb{M}}$		$A^{\mathbb{E}}$		$\text{MSE}^{\mathbb{M}}$		$\text{MSE}^{\mathbb{E}}$	
		$\mathcal{B}$	S-cube	$\mathcal{B}$	S-cube	$\mathcal{B}$	S-cube	$\mathcal{B}$	S-cube
2	0.01	<b>63.21</b>	57.12	<b>60.96</b>	56.01	<b>0.0113</b>	0.0138	<b>0.0119</b>	0.0137
	0.1	64.49	<b>64.70</b>	61.07	<b>62.22</b>	<b>0.0112</b>	0.0114	<b>0.0117</b>	0.0119
	0.5	64.41	<b>67.18</b>	61.06	<b>64.38</b>	0.0112	<b>0.0102</b>	0.0117	<b>0.0106</b>
	1	64.41	<b>67.80</b>	61.06	<b>64.71</b>	0.0112	<b>0.0102</b>	0.0117	<b>0.0108</b>
10	0.01	<b>67.91</b>	64.74	<b>68.31</b>	64.74	<b>0.0126</b>	0.0151	<b>0.0134</b>	0.0163
	0.1	68.04	<b>72.25</b>	68.36	<b>70.96</b>	0.0125	<b>0.0116</b>	0.0132	<b>0.0121</b>
	0.5	67.99	<b>73.26</b>	68.31	<b>71.72</b>	0.0125	<b>0.0109</b>	0.0132	<b>0.0112</b>
	1	68.04	<b>73.26</b>	68.31	<b>71.75</b>	0.0125	<b>0.0108</b>	0.0132	<b>0.0111</b>

**Table 1: Performance of our method S-cube compared to baseline  $\mathcal{B}$  at varying percentage of training data  $\bar{\omega}$  and style coefficient  $\alpha$ , measured in terms of (left) accuracy  $A$  and (right) mean squared error (MSE). Performances have been evaluated using both the internal  $\mathbb{M}$  and external  $\mathbb{E}$  predictor.**

## 4. EXPERIMENTAL VALIDATION

We assess the performance of our approach in successfully retrieving the most memorabilizing seeds to increase the memorability of arbitrary images (Sec. 4.2). The datasets and experimental protocol used in our study are described in Sec. 4.1.

### 4.1 Datasets and Experimental Protocol

In our experiments we consider two publicly available datasets, LaMem<sup>3</sup> and DeviantArt<sup>4</sup>.

**LaMem.** The LaMem dataset [19] is the largest dataset used to study memorability. It is a collection of 58,741 images gathered from a number of previously existing datasets, including the affective images dataset [24], which consists of Art and Abstract paintings. The memorability scores were collected for all the dataset pictures using an optimized protocol of the memorability game. The corpus was released to overcome the limitations of previous works on memorability which considered small datasets and very specific image domains. The large appearance variations of the images makes LaMem particularly suitable for our purpose.

**DeviantArt.** This dataset [29] consists of a set of 500 abstract art paintings collected from deviantArt (dA), an on-line social network site devoted to user-generated art. Since the scope of our study requires avoiding substantial modifications of the high-level content of the image, we selected the style seeds from abstract paintings. Indeed, abstract art relies in textures and color combinations, thus an excellent candidate when attempting the automatic modification of the low-level image content.

**Protocol.** In our experiments using the LaMem dataset we consider the same training (45,000 images), test (10,000 images) and validation (3,741 images) data adopted in [19]. We split the LaMem training set into two subsets of 22,500 images each (see also Section 3.3),  $\mathcal{M}$  and  $\mathcal{E}$ , which are used to train two predictors  $\mathbb{M}$  and  $\mathbb{E}$ , respectively. The model  $\mathbb{M}$  is the Scorer employed in our framework, while  $\mathbb{E}$  (which we will denote in the following as the external predictor) is used to evaluate the performance of our approach, as a proxy for human assessment. We highlight that  $\mathbb{M}$  and  $\mathbb{E}$  can be used as two independent memorability scoring functions, since  $\mathcal{M}$  and  $\mathcal{E}$  are disjoint. The validation set is used to imple-

ment the early stopping. To evaluate the performance of our scorer models  $\mathcal{M}$  and  $\mathcal{E}$ , following [19], we compute the rank correlation between predicted and actual memorability on LaMem test set. We obtain a rank correlation of 0.63 with both models, while [19] achieves a rank correlation of 0.64 training on the whole LaMem training set. As reported in [19], this is close to human performance (0.68).

The test set of LaMem (10k images) is then used (i) to learn the proposed seed Selector and (ii) to evaluate the overall framework (and the Selector in particular). In detail, we split LaMem test set into train, validation and test for our Selector with proportion 8:1:1, meaning 8,000 for training and 1,000 for validation and test. The training set for the Selector was already introduced as  $\mathcal{G}$ . We denote the test set as  $\mathcal{V}$ . The validation set is used to perform early stopping, if required.

Regarding the seeds, we estimated the memorability of all paintings of DeviantArt using  $\mathbb{M}$  and selected the 50 most and the 50 least memorable images as seeds for our study ( $\mathcal{S}$ ). The memorability scores of the deviantArt images range from 0.556 to 0.938.

**Baseline.** To the best of our knowledge this is the first work showing that it is possible to automatically increase the memorability of a generic image. For this reason, a direct and quantitative comparison with previous studies is not possible. Indeed, the recent work [19] showed that it is possible to compute accurate memorability maps from images, which can be used as bases for further image manipulations. They also observed that using a memorability map for removing image details, such as through a cartoonization process, typically lead to a memorability decrease. Oppositely, we aim to effectively increase image memorability without modifying the high level content of the images. Therefore, the approach by [19] does not directly compare with ours. The only potential competitor to our approach would be [18], except that the method is specifically designed for face photographs. Indeed, the proposed approach aims to modify the memorability while keeping other attributes (age, gender, expression) as well as the identify untouched. Therefore, the principle of [18] cannot be straightforwardly transferred to generic images. Consequently, we define an *average* baseline  $\mathcal{B}$  that consists on ranking the style seeds according to the average memorability increase for the training set, for-

<sup>3</sup><http://memorability.csail.mit.edu>

<sup>4</sup><http://disi.unitn.it/~sartori/datasets/deviantart-dataset>



mulated as:

$$\bar{m}_s^{\mathbb{M}} = \frac{1}{G} \sum_{g=1}^G m_{gs}^{\mathbb{M}}. \quad (5)$$

## 4.2 Increasing image memorability

We first evaluate the performance of our method at predicting the memorability increase of an image-seed match, where the seed is taken from the set of style seeds  $\mathcal{S}$ , and the generic image  $\mathbf{I}_v^{\mathcal{V}}$  is taken from a set of (yet) unseen images  $\mathcal{V} = \{\mathbf{I}_v^{\mathcal{V}}\}_{v=1}^V$ . We use two different performance measures: the mean squared error (MSE) and the accuracy  $A$ , which are defined as follows:

$$\text{MSE}^{\mathbb{X}} = \frac{1}{SV} \sum_{s=1}^S \sum_{v=1}^V \left( m_{vs}^{\mathbb{X}} - \mathbb{R}_s(\mathbf{I}_v^{\mathcal{V}}) \right)^2 \quad (6)$$

and

$$A^{\mathbb{X}} = \frac{1}{SV} \sum_{s=1}^S \sum_{v=1}^V (1 - |H(m_{vs}^{\mathbb{X}}) - H(\mathbb{R}_s(\mathbf{I}_v^{\mathcal{V}}))|) \quad (7)$$

where  $\mathbb{X}$  indicates the internal or external predictor, respectively  $\mathbb{X} = \{\mathbb{M}, \mathbb{E}\}$ , and  $H$  is the Heaviside step function.

Table 1 reports the performance of both the proposed approach (S-cube) and the baseline ( $\mathcal{B}$ ) under different experimental setups. Indeed, we report the accuracy (left) and the MSE (right) evaluated using the scoring model  $\mathbb{M}$  and the external scoring model  $\mathbb{E}$  (left two and right two columns of each block), for different values of  $\alpha$  and the average amount of image-seed matches  $\bar{\omega}$ . More precisely,  $\bar{\omega} = 1$  means that all image-seed pairs are used,  $\bar{\omega} = 0.1$  means that only 10% is used, and so on.

Generally speaking our method outperforms the baseline if enough image-seed pairs are available. We argue that, as it is well known, deep architectures require a *sufficient* amount of data to be effective. Indeed, when  $\bar{\omega} = 0.01$ , the network optimization procedure attempts to learn a regression from the raw image to a 100-dimensional space with, in average, only one of this dimensions propagating the error back to the network. Although this dimension is different for each image, we may be facing a situation in which not enough information is propagated back to the parameters so as to effectively learn a robust regressor. This situation is coherent when the scoring method changes from  $\mathbb{M}$  to  $\mathbb{E}$ . We can clearly observe a decrease in the performance measures when using  $\mathbb{E}$ , as expected. Indeed, since the seed selector has been trained to learn the memorability gap of  $\mathbb{M}$ , the performance is higher when using  $\mathbb{M}$  than  $\mathbb{E}$ .

Furthermore, we report the performance of our method using two different values of the style coefficient  $\alpha = \{2, 10\}$ . It can be noticed that our method performs better in terms of MSE when  $\alpha = 2$ , while accuracy is usually higher for  $\alpha = 10$ . What a priori could be seen as a divergent behavior, can be explained by the fact that imposing a higher weight to the style produces higher memorability gaps  $m_{gs}$ , thus it may generate a higher error in the estimation. We interpret these results as an indication that MSE and  $A$  can be good criteria for finding the best setup in terms of percentage of training data, but not necessarily to set other parameters.

We also investigated the impact of the network depth and trained a seed Selector using VGG16 instead of AlexNet. We fine-tuned the layers fc6, fc7, and all conv5, using Nes-torov momentum with momentum 0.9, learning rate  $10^{-3}$

$\bar{\omega}$	$A^{\mathbb{E}}$		$\text{MSE}^{\mathbb{E}}$	
	VGG16	AlexNet	VGG16	AlexNet
0.01	<b>61.56</b>	56.01	<b>0.0121</b>	0.0137
0.1	<b>64.76</b>	62.22	<b>0.0109</b>	0.0119
0.5	63.49	<b>64.38</b>	0.0111	<b>0.0106</b>
1	63.44	<b>64.71</b>	0.0111	<b>0.0108</b>

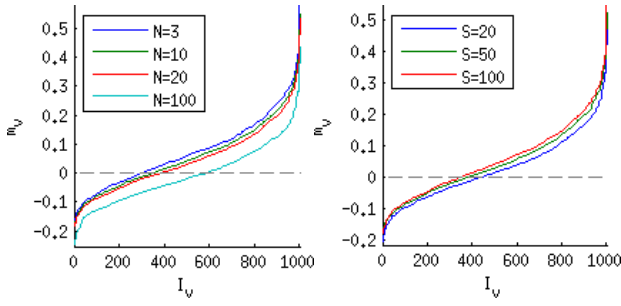
**Table 2: Performances of our method S-cube based on AlexNet (fine-tuning Hybrid-CNN [31]) and VGG16 (pre-trained on ImageNet), measured in terms of  $\text{MSE}^{\mathbb{E}}$  and  $A^{\mathbb{E}}$ , at varying percentage of training data  $\bar{\omega}$ .**

$S$	$A^{\mathbb{E}}$		$\text{MSE}^{\mathbb{E}}$	
	$\mathcal{B}$	S-cube	$\mathcal{B}$	S-cube
20	60.66	<b>63.15</b>	0.0114	<b>0.0111</b>
50	61.09	<b>63.51</b>	0.0116	<b>0.0109</b>
100	61.06	<b>64.38</b>	0.0117	<b>0.0106</b>

**Table 3: Performance of our method in terms of  $\text{MSE}^{\mathbb{E}}$  and  $A^{\mathbb{E}}$  ( $\alpha = 2$  and  $\bar{\omega} = 0.5$ ) at varying the cardinality  $S$  of the style seed set.**

and batch size 64. Importantly, while AlexNet was trained as a hybrid-CNN [19], the pre-trained model for VGG16 was trained on ImageNet. We found very interesting results and report them in Table 2, for  $\alpha = 2$ . The behavior of AlexNet was already discussed in the previous paragraphs. Interestingly we observe similar trends in VGG. Indeed, when not enough training pairs are available the results are pretty unsatisfying. However, in relative terms, the results for small  $\bar{\omega}$  are far better for VGG16 than for AlexNet. We attribute this to the fact that VGG16 is much larger, and therefore the amount of knowledge encoded in the pre-trained model has a stronger regularization effect in our problem than when using AlexNet. The main drawback is that, when enough data are available and since the amount of parameters in VGG16 is much larger than in AlexNet, the latest exhibits higher performance than the former. We recall that the seed Selector is trained with 8k images, and hypothesize that fine-tuning with larger datasets (something not possible if we want to use the splits provided in LaMem) will raise the performance of the VGG16-based seed Selector.

Furthermore, we studied the behavior of the framework when varying the size  $S$  of the seed set. Results are shown in Table 3. Specifically, we select two sets of 50 and 20 seeds out of the initial 100, randomly sampling these seeds half from the 50 most and half from the 50 least memorable ones. In terms of accuracy, the performance of both the proposed method and the baseline remain pretty stable when decreasing the number of seeds. This behavior was also observed in Table 1, especially for the baseline method. However, a different trend is observed for the MSE. Indeed, while the MSE of the proposed method increases when reducing the number of seeds (as expected), the opposite trend is found for the baseline method. We argue that, even if the baseline method is robust in terms of selecting the best seeds to a decrease of the number of seeds, it does not do a good job at predicting the actual memorability increase. Instead, the proposed method is able to select the best seeds and better



**Figure 4: Sorted average memorability gaps  $\bar{m}_v$  obtained with our method S-cube (left) averaging over varying number of top  $N$  seeds and (right) at varying the cardinality  $S$  of the seed set, with  $N = 10$ .**

measure their impact, especially when more seeds are available. This is important if the method wants to be deployed with larger seed sets. Application-wise this is quite a desirable feature since the seeds are automatically selected and hence the amount of seeds used is transparent to the user.

Finally, we assess the validity of our method as a tool for effectively increasing the memorability of a generic input image  $I_v$ . In Figure 4 (left) we report the average memorability gaps  $\bar{m}_v$  obtained over the test set  $\mathcal{V}$ , when averaging over the top  $N$  seeds retrieved, with  $N = 3, 10, 20$  and all the images. It can be noted that  $\bar{m}_v$  achieve higher values when smaller sets of top  $N$  seeds are considered, as an indication that our method effectively retrieve the most memorizable seeds. In Figure 4 (right) we report the average memorability gaps  $\bar{m}_g$  obtained over the test set  $\mathcal{V}$  with our method S-cube, considering  $N = 10$  and a varying number of style seeds  $S$ . It can be noted that a larger number of seeds allows to achieve higher increase. Figure 5 illustrates some “image memorization” sample results obtained with our method.

Summarizing, we presented an exhaustive experimental evaluation showing several interesting results. First, the proposed S-cube approach effectively learns the seeds that are expected to produce the largest increase in memorability. This increase is consistently validated when measuring it with the external scorer  $\mathbb{E}$ . We also investigated the effect of the choice of architecture for the seed Selector and the effect of the amount of seeds in the overall performance. Finally, we have shown the per-image memorability increase when using the top few seeds, and varying the size of the seed set. In all, the manuscript provides experimental evidence that the proposed method is able to automatically increase the memorability of generic images.

## 5. CONCLUSIONS

This paper presented a novel approach to increase image memorability based on the *editing-by-filtering* philosophy. Methodologically speaking, we propose to use three deep architectures as the Scorer, the Synthesizer and the Selector. The novelty of our approach relies on the fact that the Selector is able to rank the seeds according to the expected increase of memorability and select the best ones so as to feed the Synthesizer. The effectiveness of our approach both in increasing memorability and in selecting the top memorizable style seeds has been evaluated on a public benchmark.

We believe that the problem of increasing image memorability can have a direct impact in many fields like education, elderly care or user-generated data analysis. Indeed, mem-

orabilizing images could help editing educational supports, designing more effective brain training games for elderly people, producing better summaries from lifelog camera image streams or leisure picture albums.

While in this work we focused on memorability, the architecture of our approach is highly versatile and can potentially be applied to other concepts such as aesthetic judgement, virality [1] or emotions [2]. A necessary condition to this is a sufficient precision of the Scorer, which should be as close to human performance as possible. When this condition does not occur, the automatic prediction can be replaced introducing a data annotation campaign. The philosophy followed in this study could be extended to take into account other image properties such as aesthetics or evoked emotions *simultaneously*. Indeed, image manipulations aimed to increase memorability may for example lead to aesthetically unpleasing images, which could be an undesired result for an advertising campaign or a life-log video summary. This is highly interesting and not straightforward, and we consider it as one of the main future work guidelines.

While literature on predicting image abstract concepts like memorability is quite huge, the literature in image synthesis with deep networks is still in its early infancy. A promising line of work is represented by Generative Adversarial Networks (GANs) [11]. However, it is not straightforward to apply GANs and still retaining the *editing-by-filters* philosophy. Indeed, one prominent feature of our methodology is that we keep the user in the loop of the image manipulation process, by allowing them to participate to the style selection, once the most promising seeds are automatically provided. Future research works will also investigate an alternative holistic approach based on GANs.

## 6. REFERENCES

- [1] X. Alameda-Pineda, A. Pilzer, D. Xu, E. Ricci, and N. Sebe. Viraliency: Pooling local virality. In *CVPR*, 2017.
- [2] X. Alameda-Pineda, E. Ricci, Y. Yan, and N. Sebe. Recognizing emotions from abstract paintings using non-linear matrix completion. In *CVPR*, 2016.
- [3] A. K. Anderson, P. E. Wais, and J. D. Gabrieli. Emotion enhances remembrance of neutral events past. *PNAS*, 103(5):1599–1604, 2006.
- [4] A. F. Blackwell. Correction: A picture is worth 84.1 words. In *ESP Student Workshop*, 1997.
- [5] M. A. Borkin, A. A. Vo, Z. Bylinskii, P. Isola, S. Sunkavalli, A. Oliva, and H. Pfister. What makes a visualization memorable? *IEEE TVCG*, 19(12):2306–2315, 2013.
- [6] T. F. Brady, T. Konkle, G. A. Alvarez, and A. Oliva. Visual long-term memory has a massive storage capacity for object details. *PNAS*, 105(38):14325–14329, 2008.
- [7] Z. Bylinskii, P. Isola, C. Bainbridge, A. Torralba, and A. Oliva. Intrinsic and extrinsic effects on image memorability. *Vision research*, 116:165–178, 2015.
- [8] L. Cahill and J. L. McGaugh. A novel demonstration of enhanced memory associated with emotional arousal. *Consciousness and cognition*, 4(4):410–421, 1995.
- [9] L. A. Gatys, A. S. Ecker, and M. Bethge. A neural algorithm of artistic style. *arXiv preprint*



**Figure 5: Sample results: (left) original input image, (center) retrieved style seed and (right) corresponding synthesized image. The memorability score measured with the external model  $\mathbb{E}$  is reported below each image. The memorability score predicted by the Selector  $\mathbb{R}$  based on the image-seed match is reported below the resulting synthesized image.**

- arXiv:1508.06576*, 2015.
- [10] L. A. Gatys, A. S. Ecker, and M. Bethge. Image style transfer using convolutional neural networks. In *CVPR*, 2016.
  - [11] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, 2014.
  - [12] M. Gygli, H. Grabner, H. Riemenschneider, F. Nater, and L. Gool. The interestingness of images. In *ICCV*, 2013.
  - [13] M. Hilbert. How much information is there in the “information society”? *Significance*, 9(4):8–12, 2012.
  - [14] R. R. Hunt and J. B. Worthen. *Distinctiveness and memory*. Oxford University Press, 2006.
  - [15] P. Isola, D. Parikh, A. Torralba, and A. Oliva. Understanding the intrinsic memorability of images. In *NIPS*, 2011.
  - [16] P. Isola, J. Xiao, T. Antonio, and A. Oliva. What makes an image memorable? In *CVPR*, 2011.
  - [17] P. Isola, J. Xiao, D. Parikh, A. Torralba, and A. Oliva. What makes a photograph memorable? *IEEE TPAMI*, 36(7):1469–1482, 2014.
  - [18] A. Khosla, W. Bainbridge, A. Torralba, and A. Oliva. Modifying the memorability of face photographs. In *ICCV*, 2013.
  - [19] A. Khosla, A. S. Raju, A. Torralba, and A. Oliva. Understanding and predicting image memorability at a large scale. In *ICCV*, 2015.
  - [20] A. Khosla, A. D. Sarma, and R. Hamid. What makes an image popular? In *WWW*, 2014.
  - [21] A. Khosla, J. Xiao, P. Isola, A. Torralba, and A. Oliva. Image memorability and visual inception. In *SIGGRAPH Asia*, 2012.
  - [22] A. Khosla, J. Xiao, A. Torralba, and A. Oliva. Memorability of image regions. In *NIPS*, 2012.
  - [23] C. Li and M. Wand. Precomputed real-time texture synthesis with markovian generative adversarial networks. In *ECCV*, 2016.
  - [24] J. Machajdik and A. Hanbury. Affective image classification using features inspired by psychology and art theory. In *ACM Multimedia*, 2010.
  - [25] S. Maren. Long-term potentiation in the amygdala: a mechanism for emotional learning and memory. *Trends in neurosciences*, 22(12):561–567, 1999.
  - [26] J. L. McGaugh. Make mild moments memorable: add a little arousal. *Trends in cognitive sciences*, 10(8):345–347, 2006.
  - [27] K.-C. Peng, T. Chen, A. Sadovnik, and A. C. Gallagher. A mixed bag of emotions: Model, predict, and transfer emotion distributions. In *CVPR*, 2015.
  - [28] E. A. Phelps. Human emotion and memory: interactions of the amygdala and hippocampal complex. *Current opinion in neurobiology*, 14(2):198–202, 2004.
  - [29] A. Sartori, V. Yanulevskaya, A. A. Salah, J. Uijlings, E. Bruni, and N. Sebe. Affective analysis of professional and amateur abstract paintings using statistical analysis and art theory. *ACM Transactions on Interactive Intelligent Systems*, 5(2):8, 2015.
  - [30] D. Ulyanov, V. Lebedev, A. Vedaldi, and V. Lempitsky. Texture networks: Feed-forward synthesis of textures and stylized images. *ICML*, 2016.
  - [31] B. Zhou, A. Lapedriza, J. Xiao, A. Torralba, and A. Oliva. Learning deep features for scene recognition using places database. In *NIPS*. 2014.