# How to use Information Theory for Image Inpainting and Blind Spot Filling-in?

Jean-Marc Berthommé[1], Thierry Chateau[1] and Michel Dhome[1]
[1]*Institut Pascal - UMR 6602, Université Blaise Pascal, 24 Avenue des Landais, 63177 Aubière Cedex, France*
{*j-marc.berthomme, thierry.chateau, michel.dhome*}@*univ-bpclermont.fr*

Keywords: Image Inpainting, Information Theory, Blind Spot

Abstract: This paper shows how information theory can both drive the digital image inpainting process and the optical illusion due to the blind spot. The defended position is that the missing information is padded by the "most probable information around" via a simple filling-in scheme. Thus the proposed algorithm aims to keep the entropy constant. It cares not to create too much novelty as well as not to destroy too much information. For this, the image is broken down into regular squares in order to build a dictionary of unique words and to estimate the entropy. Then the occluded region is completed, word by word and layer by layer, by picking the element which respects the existing image, which minimizes the entropy deviation if there are several candidates, and which limits its potential increase in the case where no compatible word exists and where a new one must be introduced.

# 1 INTRODUCTION

The eye's blind spot has been discovered in 1660 by Edme Mariotte, a French physicist, whose experiment seemed magical when it was first presented to Louis *XIV*'s court. Today, despite three-and-a-half centuries of progress, this demonstration still resists interpretation.

From the $19^{th}$ century until now naturalists then neuroscientists have remarkably well investigated the visual system. However, despite tons of observations they do not fully understand yet its functioning. Brain modelling remains a real challenge.

On the other hand, the relatively recent community of computer vision, clearly based on hard science, has a lot to do to solve its own problems, *e.g.* segmentation, 3D reconstruction or tracking. It looks for good algorithms, not laws. As a result it has regularly claimed that it has nothing to do with medicine, perceived as too experimental.

Whatever these clichés, some pieces seem to match. Image Inpainting experiments can precisely simulate the illusion of the blind spot and more fundamentally Information Theory can *simply explain* the principle underlying the phenomenon.

# 2 THE BLIND SPOT

## 2.1 Demonstration

Let us start by showing an experiment that ophtalmologists know well. Look at the top part $(a)$ of figure 1 with the big dot and the cross. Close your right eye and force your left eye to stare at the cross slightly sidelongly. Slowly move your head closer to the screen. When the image of the dot hits your blind spot it disappears. Note that as soon as you let your left eye directly look at the dot it immediatly reappears. Then use the same distance in case $(b)$. The hole of the horizontal line is now completed so that it appears continuous. These two examples have demonstrated the existence of your blind spot.

## 2.2 Partial explanation

Neuroscientists half understand the phenomenon. They see why information is *missing* but not why it is *completed*. As shown in figure 2 it occurs where the optic nerve leaves the eye. The axons of the retinal ganglion cells concentrate at one point and go through the retina, preventing any presence of photoreceptors. In human beings the blind spot is large, about $4^o$ of the view field. It is located at slightly different angles in each eye, probably to facilitate their mutual filling. Finally, some invertebrates like cephalopods do not


Figure 1: Two experiments that demonstrate the existence of the blind spot. To make the black dot in (a) or the white hole in (b) disappear, close your right eye and force your left eye to look at the cross with a slight angle, then slowly move your head back and forth at about 25 cm from the screen.

have a blind spot. Their nerve fibers route *behind* the retina and do not block light.

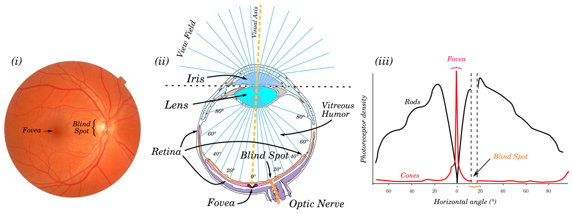


Figure 2: In the eyes of the vertebrates, the nerve fibers route *before* the retina, blocking some light and creating a blind spot where the fibers pass through the retina and out of the eye. (i) human retinography, (ii) diagram of human eye, and (iii) photoreceptor density highlight the blind spot.

## 2.3 Emulation

We are almost always unaware of our natural blind spots. So how are they naturally filled in? One radical solution would be to *experiment* with a conscious patient full of instruments in his eye and his cortex. To the best of our knowledge, this is not yet feasible. Beyond ethics, there are still technical issues.

Other people have made the parallel with Image Processing and tried to *model* the phenomenon in order to emulate the filling-in process. That is the perspective we have adopted. But, from what we read, and whatever the community, they have talked about "*visual interpolation*" (Durgin, 1995) and have applied variational approaches (Liu et al., 2007) (Arias et al., 2011) to constrained Partial Differential Equation (Satoh, 2011) (PDE) problems. We contest this interpretation. Why? To make it short, simply because interpolation can fast create unexpected things, *i.e.* disorder, in other words ***entropy***. There is a simple test that illustrates it. Let us make again experiment (b) with yellow and blue bars plus a red dot, now called experiment (c) in figure 3. Let us see if green, or any mix of the three primary colours, appears or not.

Apparently not. For us this detail is quite meaningful. Our rough guess is that when something is missing, we do not really innovate, we keep the same colours as well as the redundant patterns in presence. In the following section this idea is reformulated within the Information Theory framework. Based on that principle we describe an inpainting algorithm that aims to keep the entropy constant. It can not only reproduce the blind spot experiments (a), (b) and (c) but also complete any damaged image. We have provided MATLAB implementations able to inpaint binary, grayscale or RGB images of any bit depth at: http://wwwlasmea.univ-bpclermont.fr/Personnel/Jean-Marc.Berthomme/.

# 3 INFORMATION THEORY

## 3.1 Motivation

Information theory was born to statistically measure the uncertainty in communications. In an era of analogical technologies, Claude Shannon (Shannon, 1948) and his colleagues sought to reduce noise while improving transfer rates. Their key observation was that transmitting a noisy signal via a noisy channel still produces a noisy signal. The noisiest state, later called maximum entropy, plays the role of a fixed point. This led to the idea that *uncertainty can be bounded*, by a maximum - when the signal has reached the maximum measurable noise - and by a minimum - when the signal is certain *i.e.* fully redundant. Shannon proved that any signal $X$, whose distribution of the state is $\{p_i\}_i$, cannot be compressed beyond $H(X) = -\sum_i p_i \, log_2 \, p_i$.

John von Neumann related this to Boltzmann's works because the latter had described intensive values like temperature by an average number of collisions - extensively countable in theory - between the molecules of an ideal gas. Thus $H$ function was then called Shannon's *entropy*. As 0 K means no collision, 0 bit means no information. Similarly $H$ can only increase when information is propagated.

But the theory does not stop here. As the uncertainty of a signal can be measured, the uncertainty introduced by the channel itself can be assessed. For that, Shannon investigated the mutual information $I(X;Y)$, shared between the source signal $X$ and its copy $Y$. He showed that we cannot convey more information than the capacity $C$ of the channel, defined


Figure 3: Can green or any secondary colour turn up?

by $C = max\ I(X;Y)$ with $I(X;Y) = H(X) - H(X/Y)$. Lastly, we cannot determine if uncertainty - or information - comes from the signal or the channel.

Information Theory drove to reconsider the analogical signals in order to protect their information from noise by sampling them under discrete form. Paradoxically this quantization may involve a huge loss of information. It also led to wonder how much redundancy was necessary to recover corrupted compressed signals with error-control coding. Today its main consequence and success is the advent of our digital era.

### 3.2 Maximum entropy

The main idea behind *maximum entropy* refers to the question: "How many configurations are there?". It assumes that we are able to distinguish all of them and that we assign an equal weight, a unit by default, to each one. This implies that we work in a countable world and that any continuous feature is translated under a discrete shape. The normalization of this regular counting always drives to a discrete *uniform distribution*.

By convention people have adopted base 2 because the simplest and most interesting system gets only two observations for one state. Indeed its entropy is: $H = 1/2.log_2(2) + 1/2.log_2(2) = log_2(2) = ln(2)/ln(2) = 1$ bit, thanks to the use of $log_2$. A good interpretation of the maximum entropy - or maximum of disorder - is to consider a state whose the observations - or micro-states - can all be there. So when the system has $N$ distinct equiprobable micro-states, its entropy is: $H = log_2(N)$ as $\forall i \in [1,N], p_i = 1/N$. As soon as we leave this perfectly balanced state we necessarily favour one micro-state instead of the others. In other words it is going to be more ordered, so the entropy is going to decrease. That is the physical meaning of "Why is the maximum entropy a maximum?". But to understand "Why does the uniform distribution correspond to the maximum entropy?" we must see the mathematical proof. We do not detail it but it is a consequence of Jensen's inequality applied to the convex function $p\ logp$. It is also deeply linked to the central limit theorem.

There are several ways to sketch what is behind a single bit. As drawn in figure 4 it can be an empty piece of information with an equal chance to be one label or another one, for instance 0 or 1. But it can also be the result of a set of data whose distribution is found to be 50% - 50% by chance.

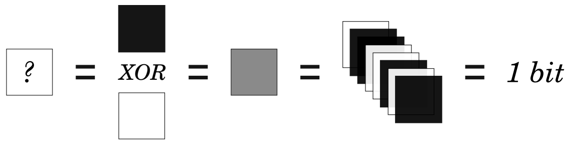


Figure 4: Examples of systems whose entropy is 1 bit.

The number of micro-states is always related to the encoding of the signal. For instance an image of $n$ binary pixels, labelled black or white, can generate $2^n$ different images. So its maximum entropy is $log_2(2^n) = n$ bits. That may look like a tautology but that is not always trivial. Indeed, as illustrated in figure 5, the way an image is broken down can highly constrain the estimation of its entropy. The state as well as the signal structure the volume of the observations. So, when talking about entropies, a good practice is to mention what is counted and what the maximum entropy reference value is.

## 4 ENTROPY INPAINTING

Inpainting is the process that replaces undesired information by contextual information without altering the global consistency of the signal. It is not necessarily restricted to image restoration as it can also apply to sounds or videos. In all cases it relies on digital signals.

### 4.1 Formulation

Inpainting should not make unexpected things arise or disappear. So, in the Information Theory framework, we state that: "Inpainting must *neither* create *nor* destroy information". By defining $X_u$, the *unknown* part of an image, and $X_k$, its *known* part, the ideal goal is to get: $H(X_u, X_k) = H(X_k)$, which is equivalent to (MacKay, 2003): $H(X_u/X_k) = 0$. Thus the proposed algorithm aims to keep the entropy of the growing known region constant. As long as there are $n_u$ unknown bits there are $2^{n_u}$ possible subimages so $H(X_u) = n_u$ bits. Concerning $H(X_k)$ its estimation requires to divide $X_k$ into pieces in order to specify a dictionary $\mathcal{D}$ and therefore propagate the filling-in process. Anyway, inpainting can be reformulated as an optimization problem looking for an unknown signal $X_u^*$ and an unknown dictionary $\mathcal{D}^*$ so that:

$$\begin{array}{ll} \text{minimize} & |H_{\mathcal{D}}(X_u, X_k) - H_{\mathcal{D}}(X_k)| \\ \text{subject to} & X_k \end{array}$$

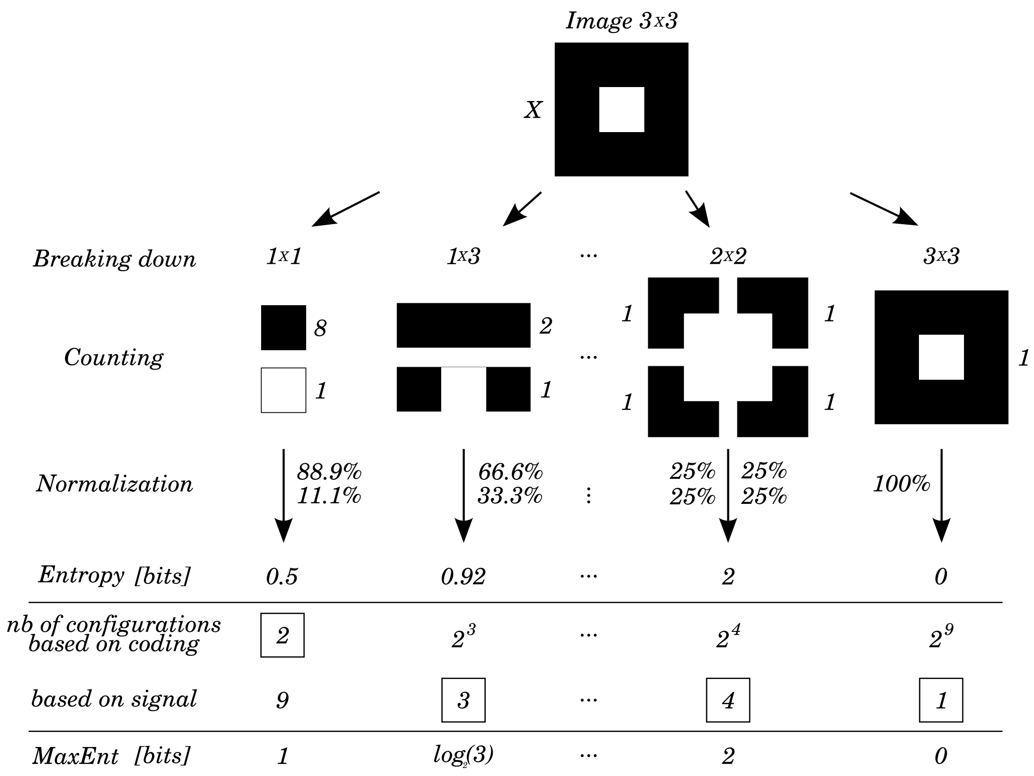


Figure 5: Breaking down of a binary image of size 3x3 into blocks of 1x1, 1x3, 2x2 and 3x3 pixels. The different dictionaries lead to different estimations of the image entropy. The maximum entropy of the image is either limited by the coding or by the signal.

## 4.2 Building the dictionary & Estimating the entropy

When the unwanted information is removed from the damaged image the method first breaks down the remaining known region. The partition possibilities are huge, from blocks of 1 x 1 pixels to the size of the image itself. Whatever the choice, the goal is to fetch the redundant patterns, so all the shapes are not relevant. Note that 1 x 1 cutting makes an exception because it loses the spatial consistency, so it is never used. Similarly, a sliding windowing is applied to the image in order not to miss any stitching between the patches. Though precise, this exhaustive enumeration is expensive. So, to shorten it, exclusive or even random windowing strategies could be considered. As shown in figure 5, the counting of the broken blocks leads to the building of a dictionary $\mathcal{D}$. It is composed of unique words associated with their frequencies $p_i$. These latter allow to estimate the entropy $H$ of the image.

## 4.3 Filling in the selection

Based on such a dictionary, any missing pixel can be replaced by looking for the patch - or the word following the point of view - that best fits the selection around. Three cases can occur. There can be one, several or no compatible words in the dictionary. Note that the compatibility is checked on the known pixels of the selection. It is calculated with the logic functions NOT, XOR and AND as each piece of image is stored as a set of Boolean. So, if there is only one word, it is always taken. If there are several, the selected one must minimize the absolute entropy deviation. Finally, if no compatible word exists a new one is created. It does not challenge the known part of the selection. It only retains the consensus within the dictionary concerning the unknown part. To conclude, the whole process is summarized in algorithm 1 below.

# 5 EXPERIMENTAL RESULTS

Our implementation has investigated dictionaries of square patches. It can emulate the blind spot completion described in section 2 with an image of size 25x100 and patches of any size between 2x2 to 25x25 in this case. The results, grouped in figure 6 below, correspond to a filling in with words of size 3x3.

We have also explored characteristic patterns like

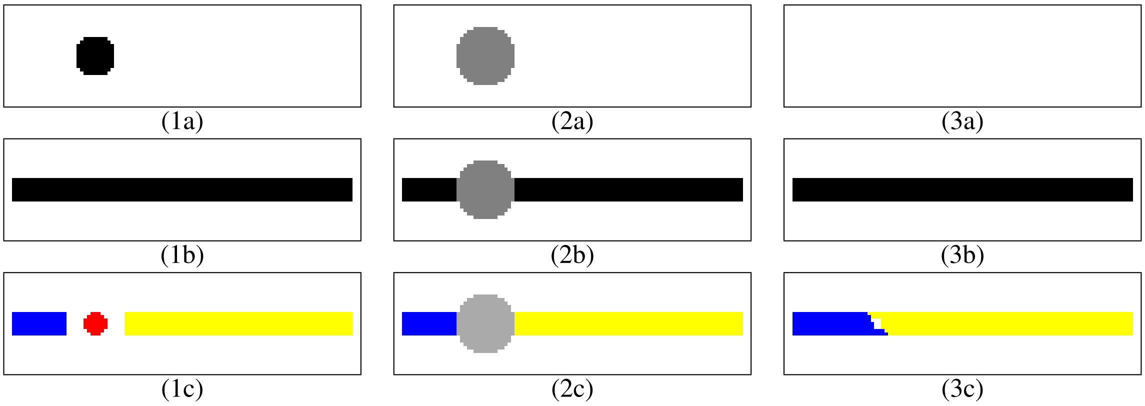


Figure 6: Entropy inpainting with 3x3 patches emulating the blind spot filling-in. (1) denotes the original images, (2) the damaged images with the same gray cache, and (3) the final inpainted images. (a), (b) and (c) refer to the experiments described in section 2.

**input** : unknown signal $X_u$ and known signal $X_k$ such that $X_u \cap X_k = \varnothing$
**output**: $X_u$ completed so that $H_{\mathcal{D}}(X_u, X_k) \approx H_{\mathcal{D}}(X_k)$

build a dictionary $\mathcal{D}$ from $X_k$ and calculate $H(X_u)$ and $H(X_k)$;
**while** *$X_u$ is not completed* **do**
    define an overlapping layer $\mathcal{L}$ between $X_u$ and $X_k$;
    **while** *$\mathcal{L}$ is not completed* **do**
        define a selection $\mathcal{S}$ inside $\mathcal{L}$;
        find the compatible words between $\mathcal{S}$ and $\mathcal{D}$;
        **if** *there is one candidate* **then**
            select this word;
        **else if** *there are several candidates* **then**
            select the word which minimizes the absolute entropy deviation;
        **else** // *there is no candidate*
            create a new word compatible with $\mathcal{S}$ and $\mathcal{D}$;
        **end**
        fill in $\mathcal{S}$ with the returned word;
        update $X_u$, $X_k$, $\mathcal{D}$ and recalculate $H(X_u)$ and $H(X_k)$;
    **end**
**end**

**Algorithm 1:** Entropy Inpainting Algorithm

crosses in order to see how to constrain the novelty creation. This work has started with binary images and was then extended to grayscale and RGB images of any bit depth. Random images have equally procured a deep reflection. We first wanted to minimize the entropy, not its absolute deviation. So we encoded redundancy and thus destroyed information. As random images are already at maximum entropy, regardless of their breaking down, we must maintain their entropy constant to be able to reproduce their pattern.

Finally we have tried to inpaint natural images taken from the benchmark dataset proposed by (Kawai et al., 2009) at: http://yokoya.naist.jp/research/inpainting. Our RMSE values compare the inpainted image to the original one within the completed region only. They are not relevant to highlight the image global consistency but they are good to compare the inpainting methods. We readily acknowledge that our implementation could highly be improved compared to the others. Clearly we do not manage edges and complex textures.

## 6 DISCUSSION

Beyond the slow speed of our algorithm, mainly due to the lack of multithreading and code optimization, our implementation most suffers from its dictionary. It only checks the redundancy of square patches of fixed size, which is strongly restricted. First, multiscale dictionaries can work like single scale ones. Second, we can fix almost no constraint on the shape of the words. The minimum constraint would be that they must be a *partition* of the image. Indeed redundant patterns can emerge at *any scale* and under *any shape*. So we must look for another way to build a good dictionary at a decent cost. Until now things have been solved in a technical way by exploring the use of edges (Liu et al., 2007) or gradients (Arias et al., 2011) or by synthesizing random textures (Harrison, 2005). These methods are very efficient but we wonder if a complete theoretical answer can be found. If there is a finite number of partitions in an image - although huge according to the Bell number - there should be a finite number of possible fillings - even huger. So among this heap of solutions one of them

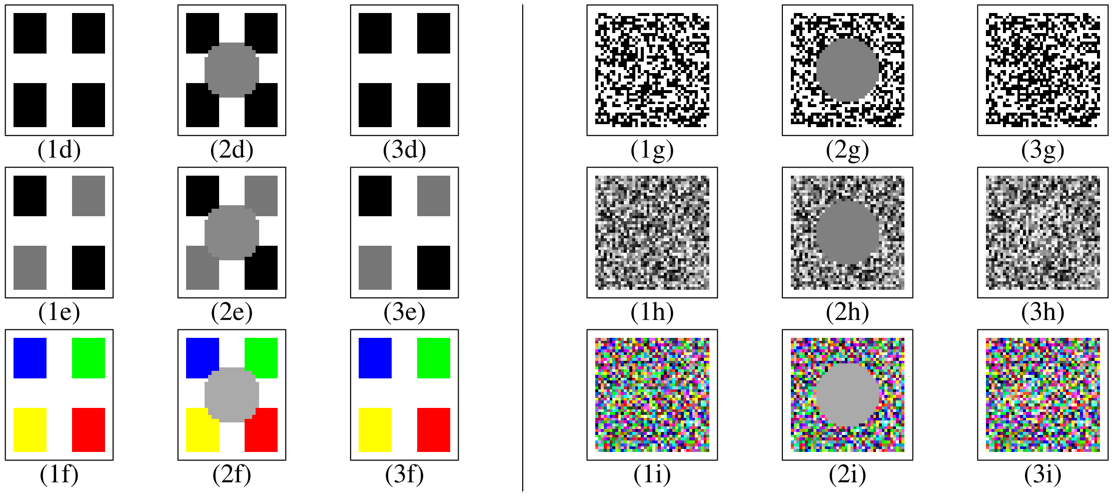


Figure 7: Entropy inpainting with 3x3 patches recovering characteristic patterns on binary, grayscale and RGB images. (1) denotes the original images, (2) the damaged images with the gray cache, and (3) the final inpainted images. (d), (e) and (f) cases are sensitive to the creation of information. (g), (h) and (i) cases are sensitive to the destruction of information.

should better respect the consistency of the image, *i.e.* induce a lower entropy.

In another perspective it might be worth *clustering* the words of the dictionary. Although it might be lossy this could avoid the dictionary dispersion and speed up the process. Similarly, it would be interesting to modulate the unknown/known pixels ratio in the selection. Our completion was *risk-averse* and slow but it could be tuned to *risk-taking* and be faster.

Concerning the compatibility between any word $\mathcal{W}$ of the dictionary and the selection $\mathcal{S}$ the criterion was *hard* with a formula looking like AND(NOT(XOR($\mathcal{S}$, $\mathcal{W}$))) and a result belonging to $\{0;1\}$. We note that it could be *soft* based on $I(\mathcal{S};\mathcal{W})/min(H(\mathcal{S}),H(\mathcal{W}))$ and with a value in $[0,1]$.

About the inpainting formulation, we wonder what links the energy to the entropy interpretation. For instance we think of the Lloyds's algorithm (Sabin and Gray, 1986), which makes any Voronoi diagram tend towards a uniform tiling, and where the two interpretations cohabit.

Finally, we remark that the sparsest dictionary implies the lowest entropy and thus the fastest filling. There is obviously a resonance with convex optimization (Boyd and Vandenberghe, 2004) like the *basis pursuit*. It solves: minimize $\sum_{i=1}^{m}(f(ui)-yi)^2+\gamma\|x_i\|_1$, where $\gamma > 0$ is a parameter used to trade off the quality of the fit to the data and the sparsity of the coefficient vector. It is now well understood that, to solve a number of problems, we both need to keep the cost function *convex*, *i.e.* use $l_p$-norms where $1 \leq p$, and get a *sparse* solution, *i.e.* use penalty functions with $l_p$-norms where $0 < p \leq 1$. Inpainting based on variational methods have almost always explored the possibilities within this framework.

# 7 CONCLUSION

This paper has underscored the fact that Information Theory can *simply formulate* the inpainting process and *precisely emulate* the blind spot filling-in. It has emphasized that the goal of inpainting is *neither* to create *nor* to destroy information. Thus the inpainting process was reformulated within the Information Theory framework under the form of an optimization problem looking for both a dictionary and an unknown signal. It aims to keep the entropy of the growing known signal constant. For that, an example of *entropy inpainting* algorithm has been proposed.

The provided implementation has simulated the described optical illusions due to the blind spot experiments. It can equally inpaint binary, grayscale or RGB images of any bit depth. However it is far for being optimal. This is mainly due to the exclusive use of fixed size square patches in the dictionary. This can be greatly improved by removing almost all the constraints on the shape of the words.

Last but not least, we are convinced that it is worth modelling simple optical illusions to push the theory to its limits and to better understand the perception process. Many things remain unclear and there are still a lot of things to do and to learn.

# REFERENCES

Arias, P., Facciolo, G., Caselles, V., and Sapiro, G. (2011). A variational framework for exemplar-based image in-

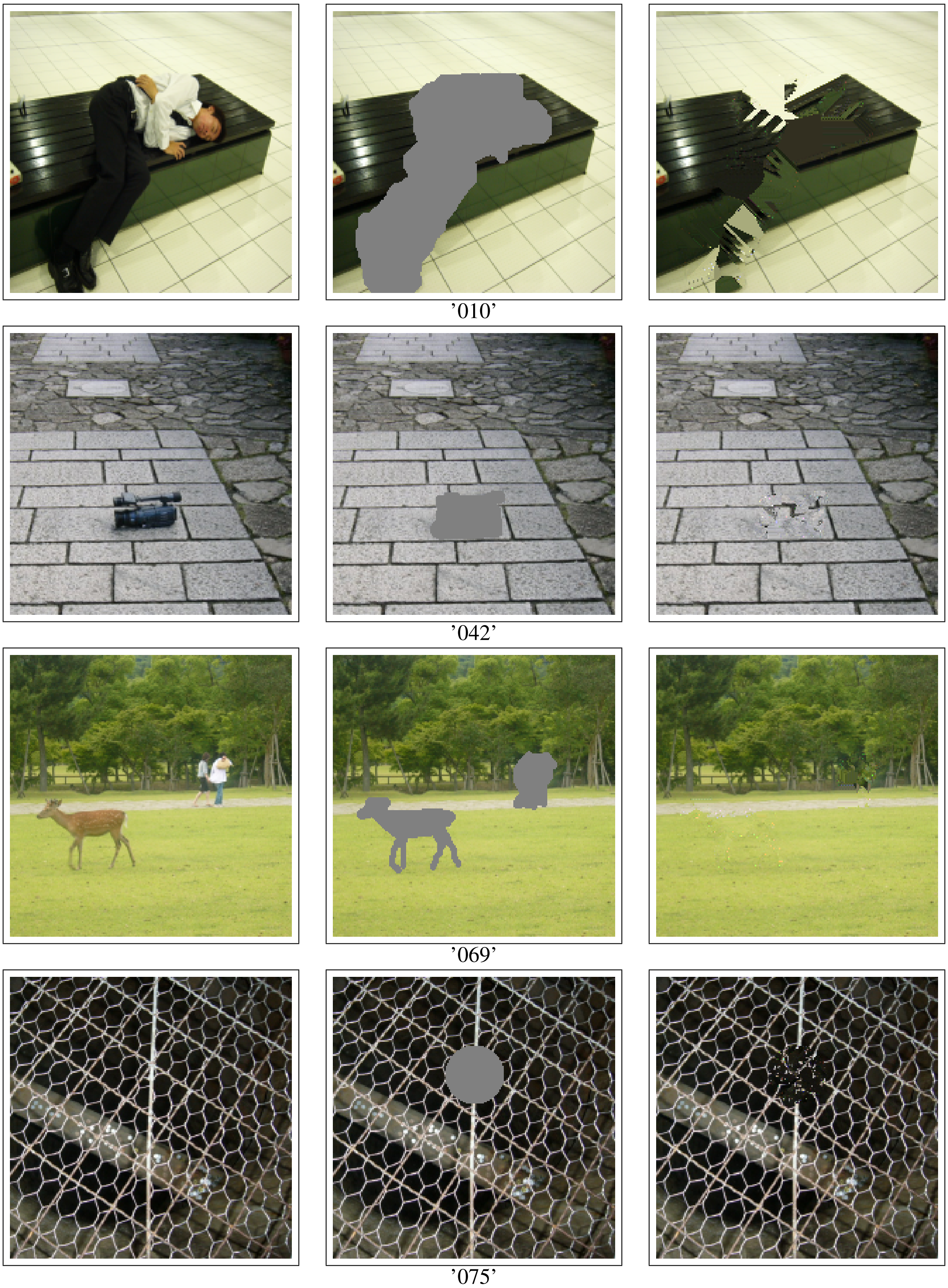


Figure 8: Entropy inpainting with 3x3 patches on natural images of size 200x200 taken from the benchmark proposed by (Kawai et al., 2009). The completions of ’010’, ’042’, ’069’ and ’075’ images have respectively RMSE of 91.7, 95.9, 62.6 and 85.1 for pixel values in $[0 - 255]^3$.

painting. *International journal of computer vision*, 93(3):319–347.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press.

Durgin, F. H. (1995). On the filling in of the visual blind spot: some rules of thumb. *Perception*, 24:827–840.

Harrison, P. F. (2005). *Image Texture Tools*. PhD thesis, Clayton School of Information Technology, Monash University.

Kawai, N., Sato, T., and Yokoya, N. (2009). Image inpainting considering brightness change and spatial locality of textures and its evaluation. *Advances in Image and Video Technology*, pages 271–282.

Liu, D., Sun, X., Wu, F., Li, S., and Zhang, Y. (2007). Image compression with edge-based inpainting. *Circuits and Systems for Video Technology, IEEE Transactions on*, 17(10):1273–1287.

MacKay, D. (2003). *Information Theory, Inference, and Learning Algorithms*. Cambridge University Press.

Sabin, M. and Gray, R. (1986). Global convergence and empirical consistency of the generalized lloyd algorithm. *IEEE Transactions on Information Theory*, 32(2):148–155.

Satoh, S. (2011). Computational identity between digital image inpainting and filling-in process at the blind spot. *Neural Computing & Applications*, pages 1–9.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 27:379–423, 623–656.