Appears in: Proc. International Conference on Computer Vision, October 2003.

Unsupervised Image Translation

Rómer Rosales, Kannan Achan, and Brendan Frey
Probabilistic and Statistical Inference Laboratory
University of Toronto, Toronto, ON, Canada
{romer,kannan,frey}@psi.toronto.edu

Abstract

An interesting and potentially useful vision/graphics task is to render an input image in an enhanced form or also in an unusual style; for example with increased sharpness or with some artistic qualities. In previous work [10, 5], researchers showed that by estimating the mapping from an input image to a registered (aligned) image of the same scene in a different style or resolution, the mapping could be used to render a new input image in that style or resolution. Frequently a registered pair is not available, but instead the user may have only a source image of an unrelated scene that contains the desired style. In this case, the task of inferring the output image is much more difficult since the algorithm must both infer correspondences between features in the input image and the source image, and infer the unknown mapping between the images. We describe a Bayesian technique for inferring the most likely output image. The prior on the output image $P(\mathcal{X})$ *is a patch-based Markov random field obtained from the source image. The likelihood of the input* $P(\mathcal{Y}|\mathcal{X})$ *is a Bayesian network that can represent different rendering styles. We describe a computationally efficient, probabilistic inference and learning algorithm for inferring the most likely output image and learning the rendering style. We also show that current techniques for image restoration or reconstruction proposed in the vision literature (e.g., image super-resolution or de-noising) and image-based nonphotorealistic rendering could be seen as special cases of our model. We demonstrate our technique using several tasks, including rendering a photograph in the artistic style of an unrelated scene, de-noising, and texture transfer.*

1 Introduction and Related Work

We pursue a formal method for modifying image statistics while maintaining image content. By appropriately manipulating image statistics, one could for example, improve image clarity or modify the image appearance into a more convenient, preferable, or useful one. For instance, one application of our approach is demonstrated in Fig. 1. We use a known painting (top) to specify the attributes that we would like to transfer to our input image (middle). Our algorithm infers the latent image (bottom), which displays the *style* specified by the painting.

Many fundamental problems in image processing are specific cases of the above problem. In image de-noising one seeks to remove *unwanted* noise from a given image to achieve a visually clearer one. In image super-resolution, given a low-resolution image, the goal is to estimate a high-resolution version of that same image. More generally, in image restoration we seek to discover what the original image looked like before it underwent the effect of some unknown (or partially known) process.

In this paper, we will maintain the scope of this problem general. Thus, we refer to our method as that of *image translation*, suggesting a similar process to that of *language translation* (despite their clear conceptual differences). Besides approaching the above *restoration* problems, our method allows, in general, to intervene in the image properties. For example, one could try to increase

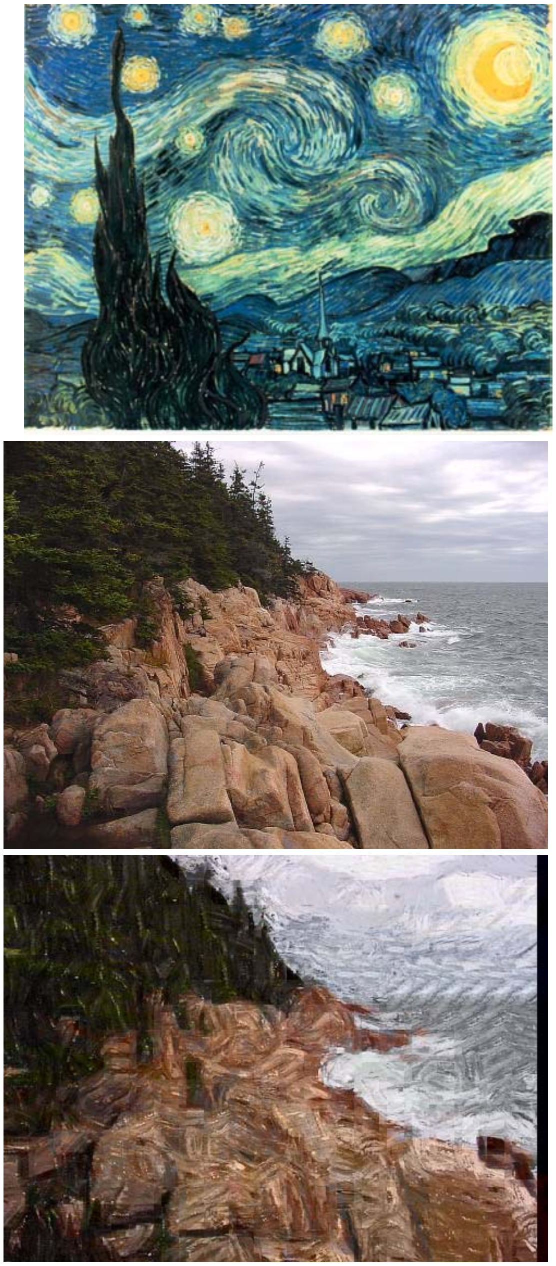

Figure 1: Image μ, *Starry Night* by van Gogh, is used to specify the source patches (top), input image $\mathcal{Y}$ (center), inferred latent image $\mathcal{X}$ (bottom).

the image photorealism (*e.g.*,relative to an originally non-photorealistic version) or change the artistic appearance of images (*e.g.*,change a painting style into another).

A variety of recent and past research work have been proposed to approach specific problems from those mentioned above, *e.g.*,[9, 26, 5, 10, 16, 23, 28, 24, 21]. From these, our approach is most closely related to [5, 26, 9] since the joint distribution of the latent image (*i.e.*,image to be estimated) is represented as a Markov Random Field (MRF) with pairwise potentials. However, there are significant differences, in our work (including the latent image representation). The primary differences pertain to:

(1) the inclusion of *unknown* transformation functions that relate (probabilistically) latent and observed image patches. Thus, the joint distribution of the model is different from that of [5, 26, 9], where we can think of these functions as known, as explained in Sec. 5. Moreover, here we allow for multiple transformation functions. We also show how these transformations can be estimated, instead of assuming that they are given.

(2) the conditional probability distribution of a patch in the latent image given a patch in the observed image cannot be estimated *directly*. In our case this distribution is *unknown*. In previous work, it was assumed that for training, an original and modified version of the source or example images were given (supervised learning). In our work we drop this assumption, we only have patches from a source image with different properties than the input image. This difference is related to the first one, since knowledge of the transformations would make this conditional probability partially (or fully) known.

(3) the derivation of new algorithms for inference. The estimation of new parameters, the use of different conditional independence assumptions, and the incorporation of new hidden random variables make estimation and inference a more complex task.

In the graphics literature, our approach is also related to [10], in the area of non-photorealistic rendering (also related to [3, 4]). In [10], a user presents two (source) images with the same content and aligned, but with two different styles. Given a new (input) image in one of the above styles, the system then transforms it into a new image displaying the other image style. A nearest neighbor algorithm is then used to match image features/pixels locally (similar to [3, 21]) and globally. Excellent results were obtained using this method. However, full supervision is required, since the user has to present two well aligned images displaying the desired relationship.

One common disadvantage of previous methods is that frequently a registered image pair is not available, but instead the user may only have the input image and also a source image of an unrelated scene that contains the appropriate style. In this case, the task of inferring the output image is much more difficult, since the algorithm must both infer correspondences between features in the input image and the source image, and infer the unknown mapping between the images. We propose a novel approach to solving this problem.

We formalize the problem and solution using a probabilistic approach, in the context of graphical models [20, 6]. The full joint distribution in our model is represented by a chain graph [15], a class of graph-represented distributions that is a superset of Bayes networks and Markov random fields [20]. In our chain graph, part of the nodes are associated to image patches which are to be inferred. Also, a set of patch transformations is used to enforce consistency in the transformation used across the image. These transformations are estimated by our algorithm, thus enabling us to discover transformations that relate the observed and estimated image.

We cast this problem into an approximate probabilistic inference/learning problem and show how it can be approached using belief propagation and expectation maximization (EM). Our image translation method is also appealing because of its generality. Most of the above approaches for image reconstruction or transformation can be seen as instances of the approach presented here, as will be discussed later.

2 Image Translation as Probabilistic Inference

We will now introduce the image translation problem. First, the problem is presented at an intuitive level. Despite several simplifications, the simple description in this section may become helpful later in the paper. We then make these ideas more precise by introducing a formal mathematical formulation.

2.1 Overview: Informal Problem Description

An intuitive way to summarize the image translation problem in the context proposed in this paper is to think of the task of finding an image $\mathcal{X}$ that satisfies certain inter-patch (*e.g.*,smoothness) and within-patch (*e.g.*,contrast) properties and that produces the observed image $\mathcal{Y}$ after every patch undergoes one of several transformations.

In contrast with previous work, we do not want to assume that we know in advance these patch transformations. Also, we will most likely not be able to satisfy *exactly* all the above properties for the new image $\mathcal{X}$. Thus, we consider a probabilistic approach where given an original image $\mathcal{Y}$, we try to construct a probable image $\mathcal{X}$ (1) which is formed by taking sample patches from a patch data-set (or dictionary), (2) whose patches are constrained to approximately satisfy certain local inter-patch properties, and (3) whose patches are *related* to corresponding patches in the original image $\mathcal{Y}$ by one or more unknown but limited patch transformations. We would also like to estimate these patch transformations, the degree of certainty we should have in these transformations for each patch, and how probable a reconstructed image is with respect to another, *e.g.*,a posterior marginal probability distribution over $\mathcal{X}$.

2.2 Definitions and Setup

We consider image representations based on a set of local, possibly overlapping, patches defined simply as a set of pixels. Let $\mathcal{Y} \in \mathcal{I}_Y$ be an image formed by a set of patches $\mathbf{y}_p$, with $p \in \mathcal{P}$, $\mathcal{P} = \{1, .., P\}$ and $\mathbf{y}_p \in \Re^S$. In this paper, S is the number of pixels in each patch (this assumes one real value per pixel; however S could also account for representations using multiple channels) [1]. We will call $\mathcal{Y}$ the input or observed image. Consider also a latent image $\mathcal{X} \in \mathcal{I}_X$, with patches $\mathbf{x}_p \in \Re^T$; $\mathcal{X}$ will be the image to be estimated, and therefore it is unknown.

[1] In general, $\mathbf{y}_p$ does not have to represent pixels, but for example, filter responses

Let $\mu \in \mathcal{I}_X$ denote a known image (or concatenation of images), here simply referred to as the source or dictionary image. Assume that the set of patches in μ are a representative sample which possesses the patch statistics that we wish $\mathcal{X}$ to display.

Consider a set of patch transformations (*i.e.*,patch translators) $\mathbf{\Lambda} = \{\Lambda_l\}_{l=\{1,...,L\}}$, where $\Lambda_l : \Re^T \rightarrow \Re^S$. In our model, the task of a single Λ_l is to transform a latent patch into an observed patch. These transformations are initially unknown, and we will try to estimate them. Estimating them intuitively accounts for discovering the set of 'painters' (or styles) that were used to paint image $\mathcal{Y}$ from an image $\mathcal{X}$ (note that we would like to achieve the inverse process). There can be multiple painters, perhaps each of them specializing in a particular image transformation. The finite random variable l_p will represent the index of the transformation employed to transform patch $\mathbf{x}_p$; $\mathbf{l} = (l_1, ..., l_P)$ is thus the vector with the indices of the patch transformations for every patch in $\mathcal{X}$.

We will consider another class of transformations, topological transformations, that can perform, for example, horizontal and vertical patch translation. These will be used as follows: a patch from μ will be moved horizontally and vertically to be positioned at a new location in image $\mathcal{X}$. This class of 2D transformations (2D translations) are defined to be the finite set of sparse permutation matrices $\mathbf{\Gamma}$, in a manner similar to [7]. In our case, each element is a matrix Γ_k of dimensions $S \times |\mu|$, with $|\mu|$ the number of pixels in the dictionary image μ. Thus, for simplicity, in the future the dictionary image μ is represented as a long 1D vector formed by stacking all the patches. We then restrict Γ_k to be a binary sparse permutation matrix of the form:

$$\begin{bmatrix} 0 & ... & 0 & 1 & 0 & 0 & 0 & ... & 0 \\ 0 & ... & 0 & 0 & 1 & 0 & 0 & ... & 0 \\ 0 & ... & 0 & 0 & 0 & 1 & 0 & ... & 0 \end{bmatrix}, \tag{1}$$

which only accounts for copying and translating a group of neighboring pixels by an integer number of pixels horizontally and vertically. This restriction is not necessary, it is straightforward to consider other classes of transformations such as rotation or shearing, using the same representation. We use the random variable t to denote the $t-th$ element of the set $\mathbf{\Gamma}$, and $\mathbf{t} = (t_1, ..., t_P)$ to represent the topological transformations for all patches in the image.

2.3 Probabilistic Model

We defined our model to have joint probability distribution represented by the chain graph of Fig. 2, which can be factorized as the product of local conditional probabilities associated with the directed edges and positive potential functions associated with the undirected edges [15, 20, 6, 14]:

$$p(\mathcal{Y}, \mathcal{X}, \mathbf{l}, \mathbf{t}|\Gamma, \Theta, \mu) = \prod_{p\in\mathcal{P}} p(\mathbf{y}_p|l_p, t_p, \Gamma, \Theta, \mu)$$
$$\prod_{p\in\mathcal{P}} P(t_p|\mathbf{x}_p) \prod_{p\in\mathcal{P}} P(l_p) \frac{1}{Z} \prod_{c\in\mathcal{C}} \psi_c(\mathbf{x}_{p\in c}), \tag{2}$$

with $\{c \in C\}$ denoting the set of latent image patches that belong to clique c in the MRF at the upper layer in Fig. 2, $\mathcal{C}$ the set of cliques in the MRF sub-graph, and ψ_c the clique potential functions.

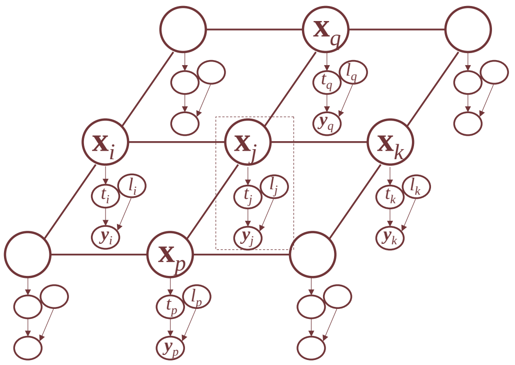

Figure 2: Chain graph representing the model joint probability distribution.

In this paper, every image patch $\mathbf{y}_p$ follows a Gaussian distribution given the patch transformation index l_p and the topological transformation t_p, with the conditional independence properties as shown in the graph in Fig. 3:

$$p(\mathbf{y}_p|l_p, t_p, \Gamma, \Theta, \mu) \quad = \quad \mathcal{N}(\mathbf{y}_p; \Lambda_{l_p}(\Gamma_{t_p}\mu), \Psi_p), \tag{3}$$

where $\Theta = \{\mathbf{\Lambda}, \mathbf{\Psi}\}$ is used to succinctly denote the distribution parameters and $\mathbf{\Psi} = \{\Psi_p\}_{p\in P}$. This is represented in Fig. 3, a sub-graph taken from the full graph in Fig. 2, representing local conditional independences for each observed image patch. For this paper, we set Λ_i to be a linear transformation[2]. This is not a restriction of our framework, it may be advantageous to also allow non-linear transformations and they could also be incorporated. However, even with linear transformations, the probabilistic model as a whole is non-linear.

In our case, we consider $p(\mathbf{x})$ to be a pairwise MRF, thus every clique c only contains two patches c_1 and c_2. Thus we simply have:

$$\psi(\mathbf{x}_c) \propto e^{d(\mathbf{x}_{c_1}, \mathbf{x}_{c_2})/2\sigma^2} \tag{4}$$

We will define d as a square distance function; thus ψ defines a Gaussian MRF. However, d is computed only on overlapping areas in the associated patches, in a way similar to [5].

In order to link our discrete transformation random variable with the continuous latent random variable $\mathcal{X}$, we use the deterministic relationship:

$$p(t_p|\mathbf{x}_p) = \begin{cases} 1 & \text{if } \mathbf{x}_p = \Gamma_{t_p}\mu \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

This is equivalent to saying that a transformation t_p will have conditional probability one, only if its patch $\mathbf{x}_p$ is equal to the patch taken from the dictionary μ using transformation t_p (here we assume that the set $\mathbf{\Gamma}$ is such that it produces unique patches).

[2] Since the inferred patches are not the result of a linear function of the observed image patches, this is different than simply transforming the image patches linearly. Instead, by this we are imposing a constraint on the *flexibility* that each of the L transformations is allowed to have.

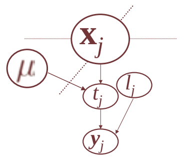

Figure 3: Sub-graph representing local conditional independences for each observed image patch. The variable μ is fixed for all patches and it is usually given in the form of the source image(s).

3 Algorithms for Learning/Inference

Let us analyze the system associated to the chain graph in Fig. 2. First we can see that it contains an undirected subgraph with loops. Even if all the model parameters Θ were known [3], inference (computing the conditional distribution over the image $\mathcal{X}$, given the image $\mathcal{Y}$) is still computationally intractable in general (in the same way as MAP estimation is). More specifically, this problem has complexity $\mathcal{O}(|\mathcal{K}|^{|\mathcal{P}|})$, with $|\mathcal{K}|$ the number of possible states that each patch $\mathbf{x}_p$ can take.

Learning the model parameters is also intractable, as can be seen from computing the derivatives of Eq. 2 with respect to the variables of interest. However, there exist approximation algorithms; one of them, based on alternating optimizations [1], is Expectation Maximization (EM). EM requires computing posterior distributions over $\mathbf{l}$ and $\mathbf{t}$, that in turn requires computing conditional marginal distributions for each node of the undirected portion of our chain graph. As we have seen, this is computationally intractable. Thus, it seems the key problem is to compute the conditional marginal distribution over the latent patches $\mathbf{x}_p$. In the following section we explain how this is done using an approximation.

Even though learning can be seen as an instance of inference, we divide this section into (1) inferring the latent image (usually referred to as inference) and (2) estimating the model parameters (usually referred to as learning).

3.1 Inference and Approximate E-step

We assume the reader have some familiarity with the EM algorithm (see *e.g.*,[18, 2]). The intractability of computing the E-step exactly can be seen as follows. In our model, the E-step is equivalent to computing the posterior:

$$P(\mathbf{l}, \mathbf{t}|\mathcal{Y}) \propto \int_{\mathcal{X}} \prod_{p \in \mathcal{P}} p(\mathbf{y}_p|l_p, t_p) P(t_p|\mathbf{x}_p) p(\mathcal{X}) d\mathcal{X}, \quad (6)$$

but we cannot solve this integral; thus we are forced to find an approximate way to perform the E-step.

Approximating patch conditional distributions. Let us say that Λ_l and Ψ_p have some value (we could initialize Λ_l and Ψ_p to a random matrix for each l and p). Then, for every p we can select the set $\mathcal{K}_p$ of K most likely topological transformations given the dictionary μ. This can be easily done for each patch in $\mathcal{X}$ once $P(t_p, l_p|y_p)$ (see below) is computed by taking those topological transformations with highest probability per hidden image patch. This takes $\mathcal{O}(TL)$ probability evaluations per patch and a sorting operation among T elements. The approximation is necessary for computational reasons and can be made as exact as desired if we are willing to pay the associated computational cost. This approximation was also used in a similar way in [5]; it accounts for cutting off the *long tails* of the distribution $P(t_p|\mathbf{y}_p)$ (or the approximation to $P(t_p|\mathcal{Y})$) by ignoring very unlikely topological transformations.

Using this succinct representation, we can then compute an approximate E-step. This is equivalent to inferring the image patches $\mathbf{x}_p$ given the parameters so far estimated and the current posterior distributions over t_p. Note that (1)$\mathcal{X}$ is conditionally independent of the rest of the model variables given $\mathbf{t}$ and (2) we can compute the marginal-conditional distributions over t_p alone by simply using $P(t_p|\mathbf{y}_p) = \sum_{l_p} P(t_p, l_p|\mathbf{y}_p)$.

Inferring the latent image. We have reduced our problem to that of inferring a distribution over the states of $\mathcal{X}$ given a distribution over t_p for all p. One way to perform this computation is by performing *loopy* belief propagation in the MRF for $\mathcal{X}$ to compute the posterior marginals over each $p(\mathbf{x}_p)$. Loopy belief propagation accounts for approximating $p(\mathbf{x}_p)$ by using the belief propagation message passing updates $m_{i\to j}$ [20] for several iterations, which in our model can be formally written as follows:

$$\begin{aligned} m_{i\to j}(\mathbf{x}_j) &= \sum_{\mathbf{x}_i = s_i} P(\mathbf{x}_i|t_i)\psi_{ij}(\mathbf{x}_i, \mathbf{x}_j) \prod_{k \in \mathcal{N}(i)\setminus j} m_{k\to i}(\mathbf{x}_i) \\ b_i(\mathbf{x}_i) &= P(\mathbf{x}_i|t_i) \prod_{k \in \mathcal{N}(i)} m_{k\to i}(\mathbf{x}_i), \end{aligned}$$

with $\mathcal{N}(i)$ the neighbors and s_i the candidates for $\mathbf{x}_i$.

These updates guarantee that upon convergence the marginal probabilities $p(\mathbf{x}_i|\mathcal{Y})$, obtained by simply normalizing $b_i(\mathbf{x}_i)$, would be at least at a local minimum of the corresponding *Bethe* free energy [27] of the (conditional) MRF. The domain of $\mathbf{x}_p$ is in practice discrete since the probability distribution is concentrated only at the candidate patches given by $\mathcal{K}_{lp}$. Thus, every full iteration has complexity $\mathcal{O}(K^2)$. With knowledge of the (approximate) conditional marginals, the E-step from Eq. 6 is then given by:

$$P(t_p, l_p|y_p) \propto \sum_{\mathbf{x}_p} p(\mathbf{x}_p|\mathcal{Y}) P(t_p|\mathbf{x}_p) P(l_p) p(\mathbf{y}_p|l_p, t_p). \quad (7)$$

Once this is done, we can then perform the M-step (as explained next) and iterate the EM algorithm as usual. Even though loopy belief propagation is not exact (clearly, since this problem cannot be solved exactly) and not guaranteed to converge, some recent work supports this approximation [8, 13, 17]. Other approximations have also been suggested (*e.g.*,[25, 12, 9]). In our case loopy belief propagation seem to provide accurate posterior marginals for the experiments performed next.

3.2 Learning the Translation Parameters

Once we performed the E-step, the M-step optimizes the expected value of the model joint distribution under $P(t_p, l_p|y_p)$ with respect to the model parameters Λ_l and Ψ_p. This can be done by computing first derivatives, as in a MAP estimate setting. For linear transformation functions, we can obtain a closed-form solution for the optimal

[3] more importantly, if the transformation indices t_p were known.

values of the parameters:

$$\Lambda_l = \frac{\sum_p \sum_{t_p} P(t_p, l_p = l|y_p) y_p (\Gamma_{t_p}\mu)^\top}{\sum_p \sum_{t_p, l_p} P(t_p, l_p|y_p)(\Gamma_{t_p}\mu)(\Gamma_{t_p}\mu)^\top} \quad (8)$$

$$\Psi_p = \frac{1}{Z_p} \sum_{t_p} \sum_{l_p} P(t_p, l_p|y_p) (y_p - \Lambda_{l_p}(\Gamma_{l_t}\mu))(y_p - \Lambda_{l_p}(\Gamma_{l_t}\mu))^\top, \quad (9)$$

with $Z_p = \sum_{t_p} \sum_{l_p} P(t_p, l_p|y_p)$.

For non-linear transformation functions, we need to use non-linear function optimization, such as gradient methods. In any case, the gradient can be computed efficiently.

In summary, learning/inference can be done exactly only up to inferring the marginal distribution of the latent patches (a problem which is in NP). One way around this problem is to approximate these marginal distributions and use them to update the model parameters.

4 Experimental Results

We illustrate the image translation method in diverse tasks. Using our method, these tasks can all be seen as instances of the same problem. In each task, between four and seven transformations L were chosen. The dimensionality of the latent and observed image patches has been reduced in dimensionality by 25% using Principal Component Analysis. This is mainly of numerical concern since even a 20×20 patch will generate a vector of dimensionality 400, for which it is hard to compute statistics using finite precision computations and small datasets. All the patches considered in these experiments are square patches. The overlap between neighboring patches used in Eq. 4 to compute the clique potentials is set to four pixels deep. We use the luminance value (from the YIQ color space) as our representation for each pixel (instead of its RGB values) [10]. For color images, the color components (IQ) are then simply copied to the final estimated image. Most images can be (much) better perceived directly from the computer screen due to resolution / space limitations (these and more tests are also available at `http://www.psi.toronto.edu`).

4.1 De-blurring / De-noising

In our first experiment, we strongly down-sampled a photographic image as seen in Fig. 4(left). This will be our observed image $\mathcal{Y}$. We used four transformations and 15×15 patches for both observed and latent image. Our goal is to obtain a higher-resolution version of the observed image. Thus, as our dictionary, we used photographic patches with the desired resolution (female face photos). The result is shown in Fig. 4(right). The system was able to infer the high-resolution patches significantly well given the information present in the considerably degraded input image (also, see video for MAP estimates at each iteration). In previous work's experimental evaluation, it is rare to see tests performed with this level of degradation in the input image. Note that our model is meant to solve the more general task of learning arbitrary convolution kernels *i.e.*,not just de-blurring kernel.

4.2 Photo-to-Line-Art Translation

Now the goal is to make our input image acquire the line-art attributes of an unrelated source image. For this task we used patches of size 15×15 and $L = 5$. We chose three source examples, shown in Fig. 5, and applied them to the image in Fig. 6(left). The results show that our method is suitable for performing this task also. On the positive side, note that even larger scale properties are accurately displayed by the corresponding inferred images; this provides excellent line consistency across the image. On the negative side, the algorithm has some difficulty in regions with strong depth discontinuities, such as the outer face contour, perhaps because it does not count with patches with enough detail in the source image.

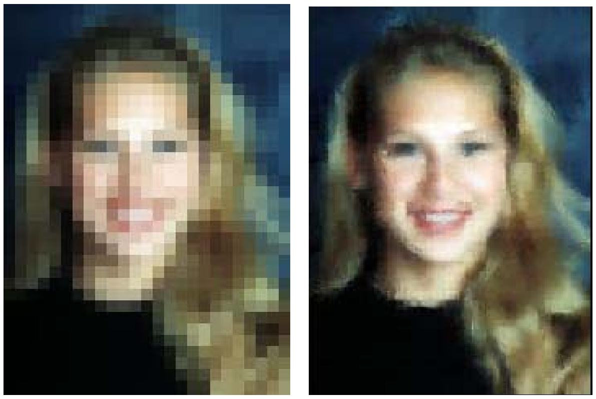

Figure 4: Input image (left) and inferred image (MAP)(right).

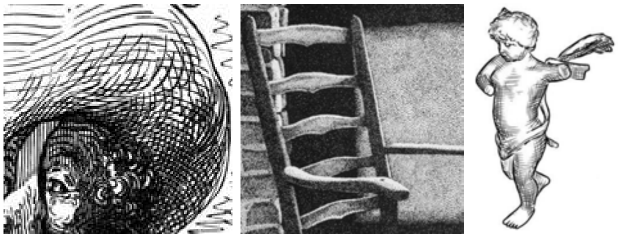

Figure 5: Source line art examples: engraving from Gustave Doré's illustrations of *Don Quixote*, stippling by Claudia Nice[19], and *Cupid* [11].

4.3 Photo-to-Paint Translation

We now apply renowned artistic styles to the input images. We use an image of a well-known painting by van Gogh, shown in Fig. 1(top), as the image with the desired attributes, and the photo in Fig. 1(center) as the input. We used L=4 and a 30×30 patch size. The inferred image in Fig. 1(bottom) inherited the local patch statistics and also global features found in the source painting. We repeated the experiment using the same painting but different input image; results are shown in Fig. 7. This image also acquired the source style. However, Lena's eyes could not be inferred with enough accuracy (or pleasing artistic detail), perhaps also because of the lack of patches with the correct properties in the original painting. We also employed paintings with different properties, on example is shown in Fig. 8. In order to more carefully observe the details in the image translation, zoomed-in areas of previous examples are shown in Fig. 9.

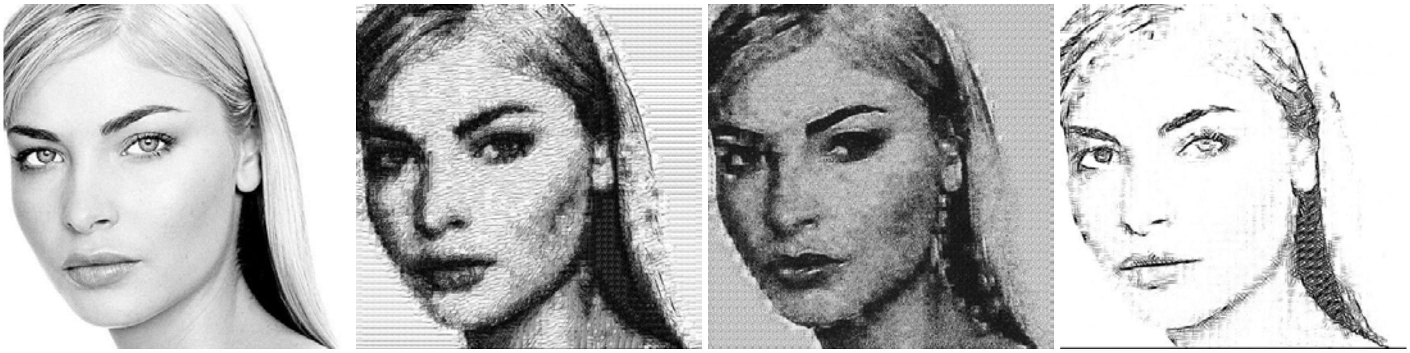

Figure 6: Input image $\mathcal{Y}$ (left) and inferred images using the three source line art example images in Fig. 5 (in same order)

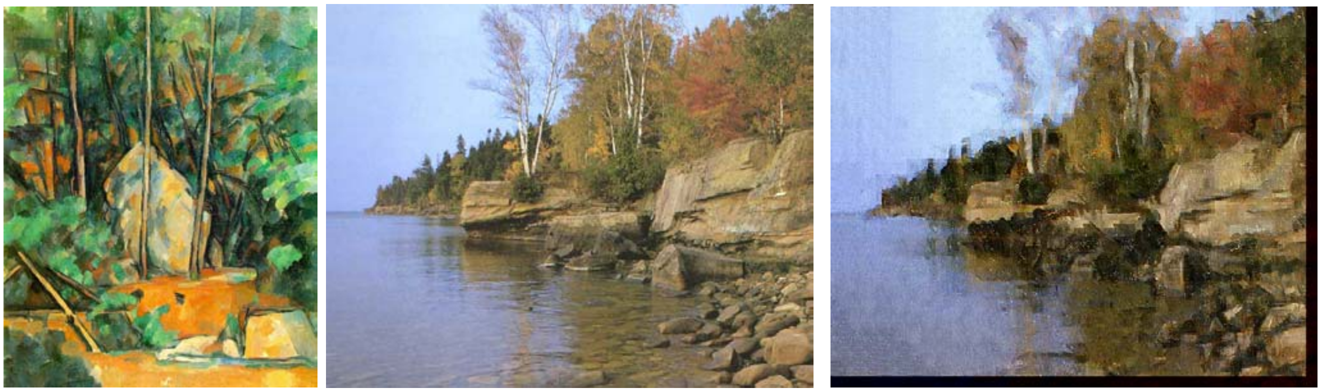

Figure 8: Source image μ, *Cistern in the Park at Chateau Noir* by Cezanne (left), input image $\mathcal{Y}$, shore photo by John Shaw[22] (center), inferred image $\mathcal{X}$ (right).

5 Relation to Previous Methods

We now briefly compare our model with several relevant approaches, proposed to solve more specific problems. The model in [5], was proposed to perform image super-resolution. Using our framework, this approach can be obtained by (1) setting $L = 1$, *i.e.*,use only one transformation and (2) Λ_1 equal to a fixed and known low-pass filter; in [5] there is no need to estimate Λ because it is given. Another important difference is that the input to this algorithm is different than the input to our algorithm. This model assumes that there exist a set of image pairs for training. Each pair consists of the high and the low resolution version of an image (the low-res version can be simply generated using the low-pass filter transformation). Our algorithm does not assume that we have access to these image pairs for training, but that we only have access to one image with the desired statistics. This a much more difficult computational and modeling task which also has important practical implications: it is more restrictive to have to find a pair of perfectly aligned images, one with the desired statistics (*e.g.*,style) and the other one obtained in the same mode as the input image we hold.

The model presented in [10], proposed for non-photorealistic rendering is also, like [5], a model where it is assumed that there exist a set of image pairs for training, with the statistics of $\mathcal{Y}$ and $\mathcal{X}$. Λ is not given, but in the supervised approach taken in this work, Λ can be easily found. More specifically, Λ is the equivalent of a nearest neighbor algorithm, which is easy to compute (given the training image pair). Large scale consistency was achieved using also a nearest neighbor approach, which in our model is equivalent to learning the distribution over $\mathcal{X}$ from our sample images and using it to define the MRF energy function (*i.e.*,effectively replacing Eq. 4)

Inference in [9] can be seen as inference in our model using only the MRF sub-graph, with observations $\mathbf{y}_p$ directly linked to unobserved nodes $\mathbf{x}_p$. Also, instead of belief propagation, in [9] Monte Carlo techniques were proposed to infer the hidden state of the image patches $\mathbf{x}_p$.

6 Conclusions

The image translation approach proposed here provides a general formalism for the analysis of a variety of problems in image processing where the goal is to estimate an unobserved image from another (observed) one. These include a number of fundamental problems previously approached using separate methods. In this sense, the image translation approach can be of practical and theoretical interest. Several practical extensions are possible, for example in the choice of a different approximate inference method, in the choice of clique potential functions used in the MRF, or in the form of the patch transformations. These changes are likely to be application dependent, and can be easily incorporated within the framework presented here.

Acknowledgments

We thank Aaron Hertzmann for helpful discussions on non-photorealistic rendering and for evaluation of results.

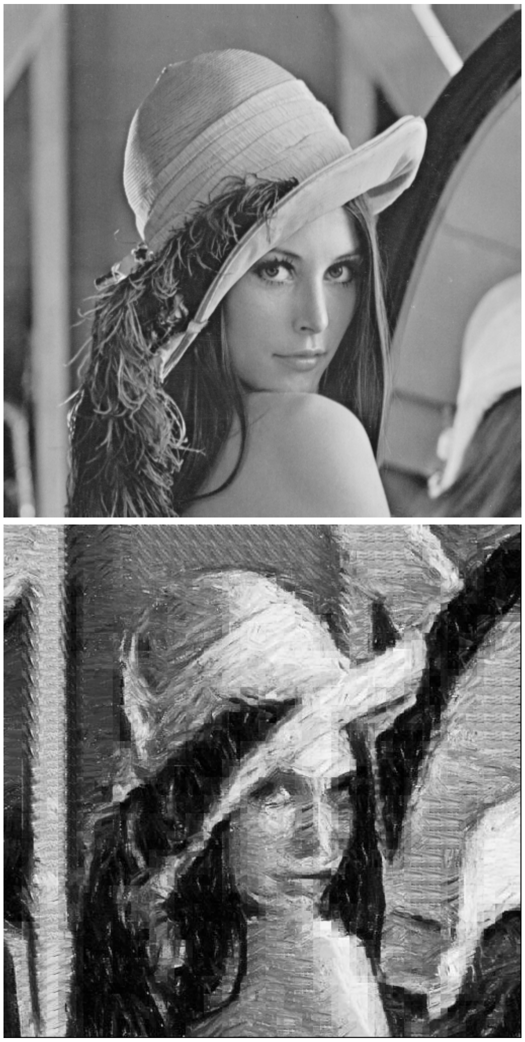

Figure 7: Input image $\mathcal{Y}$ (top), inferred latent image $\mathcal{X}$ (bottom).

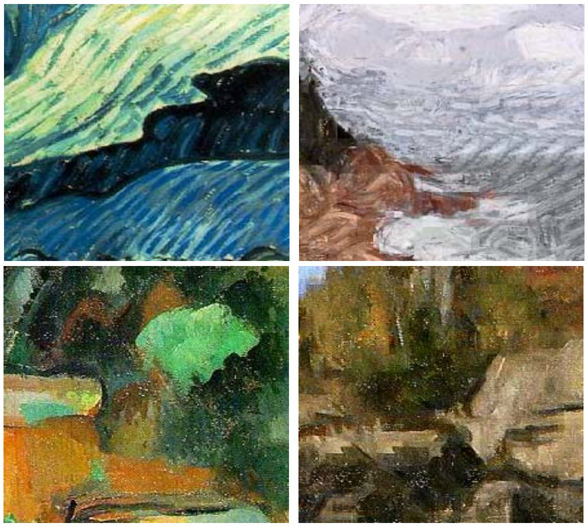

Figure 9: Two zoomed-in translations from previous examples: source image (real painting) (left) and inferred image (right); input images (not shown here) were the example landscape photos.

References

[1] I. Csiszar and G. Tusnady. Information geometry and alternating minimization procedures. *Statistics and Decisions*, 1:205–237, 1984.

[2] A. Dempster, N. Laird, and D. Rubin. Maximum likelihood estimation from incomplete data. *Journal of the Royal Statistical Society (B)*, 39(1), 1977.

[3] A. Efros and W. Freeman. Quilting for texture synthesis and transfer. In *International Conference on Computer Vision*, 1999.

[4] A. Efros and W. Freeman. Texture synthesis by non-parametric sampling. In *SIGGRAPH*, 2001.

[5] B. Freeman, E. Pasztor, and O. Carmichael. Learning low-level vision. *IJCV*, 2000.

[6] B. Frey. *Graphical Models for Machine Learning and Digital Communication*. MIT Press, 1998.

[7] B. Frey and N. Jojic. Learning graphical models of images, videos and their spatial transformations. In *Uncertainty in Artificial Intelligence*, 2000.

[8] B. Frey and D. MacKay. A revolution: Belief propagation in graphs with cycles. In *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.

[9] D Geman and S Geman. Stochastic relaxation, gibbs distribution and bayesian restoration of images. *IEEE Trans. PAMI*, 6(6):721–741, 1984.

[10] A. Hertzmann, C. Jacobs, N. Oliver, B. Curless, and D. Salesin. Image analogies. http://www.mrl.nyu.edu/projects/image-analogies/. In *SIGGRAPH 2001.*, pages 327–340, 2001.

[11] A. Hertzmann and D. Zorin. Illustrating smooth surfaces. In *SIGGRAPH 2000.*, pages 517–526, 2001.

[12] M. Jordan, Z. Ghahramani, T.Jaakkola, and L. Saul. An introduction to variational methods for graphical models. *Learning in Graphical Models, M. Jordan (editor)*, 1998.

[13] F. Kschischang and B. Frey. Iterative decoding of compound codes by probability propagation in graphical models. *IEEE J. Selected Areas in Communications*, 16:219–230, 1998.

[14] F. Kschischang, B. Frey, and H. Loeliger. Factor graphs and the sum-product algorithm. *IEEE Transactions on Information Theory*, 2001.

[15] S. L. Lauritzen and N. Wermuth. Graphical models for associations between variables, some of which are qualitative and some quantitative. *Annals of Statistics*, 17:31–57, 1989.

[16] S. Li. *Markov Random Field Modeling in Computer Vision*. Springer-Verlag, 1995.

[17] R. McEliece, D. MacKay, and J. Cheng. Turbo decoding as an instance of pearl's belief propagation algorithm. *IEEE J. Selected Areas in Communications*, 16, 1998.

[18] R. Neal and G. Hinton. A view of the em algorithm that justifies incremental, sparse, and other variants. *Learning in Graphical Models, M. Jordan (editor)*, 1998.

[19] C. Nice. *Sketching Your Favorite Subjects in Pen and Ink*. F-W Publications, 1993.

[20] J. Pearl. *Probabilistic Reasoning in Intelligent Systems*. Morgan-Kaufman, 1988.

[21] A. Pentland and B. Horowitz. A practical approach to fractal-based image compression. *Digital Images and Human Vision*, 1993.

[22] F. Petrie and J. Shaw. *The Big Book of Painting Nature in Watercolor*. Watson-Guptill, 1990.

[23] J. Portilla and E. Simoncelli. Statistics of complex wavelet coefficients. *International Journal of Computer Vision*, 40(1), 2000.

[24] R. Schultz and R. Stevenson. A bayesian approach to image expansion for improved definition. *IEEE Transactions on Image Processing*, 3(3):233–242, May 1994.

[25] M. Wainwright, T. Jaakkola, and A. Willsky. A new class of upper bounds on the log partition function. In *Uncertainty in Artificial Intelligence*, 2002.

[26] Y. Weiss. Interpreting images by propagating bayesian beliefs. In *Advances in Neural Information Processing Systems*, volume 9, page 908, 1997.

[27] J. Yedidia. An idiosyncratic journey beyond mean field theory. *Advanced Mean Field Methods*, pages 21–36, 2001.

[28] S. Zhu and D. Mumford. Prior learning and gibbs reaction-diffusion. *IEEE Trans. PAMI*, 1997.