Structural Consistency and Controllability for Diverse Colorization

Safa Messaoud, David Forsyth, and Alexander G. Schwing

University of Illinois at Urbana-Champaign, USA

Abstract. Colorizing a given gray-level image is an important task in the media and advertising industry. Due to the ambiguity inherent to colorization (many shades are often plausible), recent approaches started to explicitly model diversity. However, one of the most obvious artifacts, structural inconsistency, is rarely considered by existing methods which predict chrominance independently for every pixel. To address this issue, we develop a conditional random field based variational auto-encoder formulation which is able to achieve diversity while taking into account structural consistency. Moreover, we introduce a controllability mechanism that can incorporate external constraints from diverse sources including a user interface. Compared to existing baselines, we demonstrate that our method obtains more diverse and globally consistent colorizations on the LFW, LSUN-Church and ILSVRC-2015 datasets.

Keywords: Colorization, Gaussian-Conditional Random Field, VAE

1 Introduction

Colorization of images requires to predict the two missing channels of a provided gray-level input. Similar to other computer vision tasks like monocular depth-prediction or semantic segmentation, colorization is ill-posed. However, unlike the aforementioned tasks, colorization is also ambiguous, i.e., many different colorizations are perfectly plausible. For instance, differently colored shirts or cars are very reasonable, while there is certainly less diversity in shades of façades. Capturing these subtleties is a non-trivial problem.

Early work on colorization was therefore interactive, requiring some reference color image or scribbles [1–6]. To automate the process, classical methods formulated the task as a prediction problem [7, 8], using datasets of limited sizes. More recent deep learning methods were shown to capture more intricate color properties on larger datasets [9–14]. However, all those methods have in common that they only produce a single colorization for a given gray-level image. Hence, the ambiguity and multi-modality are often not modeled adequately. To this end, even more recently, diverse output space distributions for colorization were described using generative modeling techniques such as variational auto-encoders [15], generative adversarial nets [16], or auto-regressive models [17, 18].

While approaches based on generative techniques can produce diverse colorizations by capturing a dataset distribution, they often lack structural con-
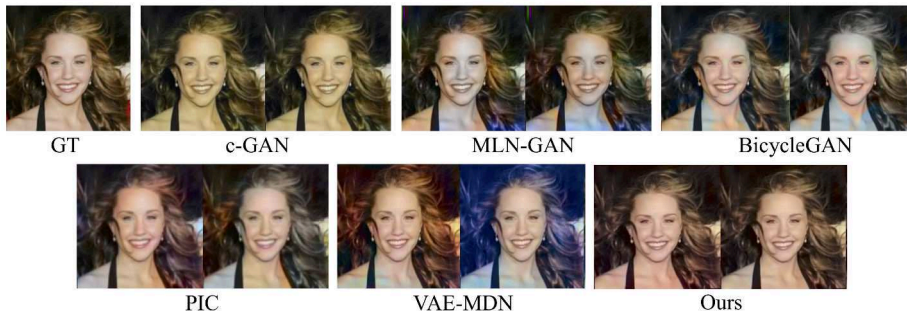
Fig. 1: Diverse colorizations of the ground truth (GT) generated by c-GAN [16], MLN-GAN [19], BicycleGAN [20], PIC [18], VAE-MDN [15] and our approach.

sistency, e.g., parts of a shirt differ in color or the car is speckled. Inconsistencies are due to the fact that structural coherence is only encouraged implicitly when using deep net based generative methods. For example, in results obtained from [15, 16, 18–20] illustrated in Fig. 1, the color of the shoulder and neck differ as these models are sensitive to occlusion. In addition, existing diverse colorization techniques also often lack a form of controllability permitting to interfere while maintaining structural consistency.

To address both consistency and controllability, our developed method enhances the output space of variational auto-encoders [21] with a Gaussian Markov random field formulation. Our developed approach, which we train in an end-to-end manner, enables explicit modeling of the structural relationship between multiple pixels in an image. Beyond learning the structural consistency between pixels, we also develop a control mechanism which incorporates external constraints. This enables a user to interact with the generative process using color stokes. We illustrate visually appealing results on the Labelled Faces in the Wild (LFW) [22], LSUN-Church [23] and ILSVRC-2015 [24] datasets and assess the photo-realism aspect with a user study.

2   Related Work

As mentioned before, we develop a colorization technique which enhances variational auto-encoders with Gaussian Markov random fields. Before discussing the details, we review the three areas of colorization, Gaussian Markov random fields and variational auto-encoders subsequently.

Colorization: Early colorization methods rely on user-interaction in the form of a reference image or scribbles [1–6]. First attempts to automate the colorization process [7] rely on classifiers trained on datasets containing a few tens to a few thousands of images. Naturally, recent deep net based methods scaled to much larger datasets containing millions of images [9–14, 25]. All these methods

operate on a provided intensity field and produce a single color image which doesn't embrace the ambiguity of the task.

To address ambiguity, Royer *et al.* [18] use a PixelCNN [26] to learn a conditional model $p(\boldsymbol{x}|\boldsymbol{g})$ of the color field $\boldsymbol{x}$ given the gray-level image $\boldsymbol{g}$, and draw multiple samples from this distribution to obtain different colorizations. In addition to compelling results, failure modes are reported due to ignored complex long-range pixel interactions, e.g., if an object is split due to occlusion. Similarly, [17] uses PixelCNNs to learn multiple embeddings $\boldsymbol{z}$ of the gray-level image, before a convolutional refinement network is trained to obtain the final image. Note that in this case, instead of learning $p(\boldsymbol{x}|\boldsymbol{g})$ directly, the color field $\boldsymbol{x}$ is represented by a low dimensional embedding $\boldsymbol{z}$. Although, the aforementioned PixelCNN based approaches yield diverse colorization, they lack large scale spatial coherence and are prohibitively slow due to the auto-regressive, i.e., sequential, nature of the model.

Another conditional latent variable approach for diverse colorization was proposed by Deshpande *et al.* [15]. The authors train a variational auto-encoder to produce a low dimensional embedding of the color field. Then, a Mixture Density Network (MDN) [27] is used to learn a multi-modal distribution $p(\boldsymbol{z}|\boldsymbol{g})$ over the latent codes. Latent samples are afterwards converted to multiple color fields using a decoder. This approach offers an efficient sampling mechanism. However, the output is often speckled because colors are sampled independently for each pixel.

Beyond the aforementioned probabilistic formulations, conditional generative adversarial networks [16] have been used to produce diverse colorizations. However, mode collapse, which results in the model producing one color version of the gray-level image, is a frequent concern in addition to consistency. This is mainly due to the generator learning to largely ignore the random noise vector when conditioned on a relevant context. [19] addresses the former issue by concatenating the input noise channel with several convolutional layers of the generator. A second solution is proposed by [20], where the connection between the output and latent code is encouraged to be invertible to avoid many to one mappings. These models show compelling results when tested on datasets with strong alignment between the samples, e.g., the LSUN bedroom dataset [23] in [19] and image-to-image translation datasets [28–32] in [20]. We will demonstrate in Sec. 4 that they lack global consistency on more complex datasets.

In contrast to the aforementioned formulations, we address both diversity and global structural consistency requirements while ensuring computational efficiency. To this end we formulate the colorization task by augmenting variational auto-encoder models with Gaussian Conditional Random Fields (G-CRFs). Using this approach, beyond modeling a structured output space distribution, controllability of the colorization process is natural.

Gaussian Conditional Markov Random Field: Markov random fields [33] and their conditional counter-part are a compelling tool to model correlations between variables. Theoretically, they are hence a good match for colorization tasks where we are interested in reasoning about color dependencies between

different pixels. However, inference of the most likely configuration in classical Markov random fields defined over large output spaces is computationally demanding [34–37] and only tractable in a few special cases.

Gaussian Markov random fields [38] represent one of those cases which permit efficient and exact inference. They model the joint distribution of the data, e.g., the pixel values of the two color channels of an image as a multi-variate Gaussian density. Gaussian Markov random fields have been used in the past for different computer vision applications including semantic segmentation [39–41], human part segmentation and saliency estimation [40, 41], image labeling [42] and image denoising [43, 44]. A sparse Gaussian conditional random field trained with a LEARCH framework has been proposed for colorization in [8]. Different from this approach, we use a fully connected Gaussian conditional random field and learn its parameters end-to-end with a deep net. Beyond structural consistency, our goal is to jointly model the ambiguity which is an inherent part of the colorization task. To this end we make use of variational auto-encoders.

Variational Auto-Encoders: Variational auto-encoders (VAEs) [21] and conditional variants [45], i.e., conditional VAEs (CVAEs), have been used to model ambiguity in a variety of tasks [46, 47]. They are based on the manifold assumption stating that a high-dimensional data point $\boldsymbol{x}$, such as a color image, can be modeled based on a low-dimensional embedding $\boldsymbol{z}$ and some auxiliary data $\boldsymbol{g}$, such as a gray-level image. Formally, existence of a low-dimensional embedding space and a transformation via the conditional $p_\theta(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{g})$ is assumed. Given a dataset $\mathcal{D}$ containing pairs of conditioning information $\boldsymbol{g}$ and desired output $\boldsymbol{x}$, i.e., given $\mathcal{D} = \{(\boldsymbol{g}, \boldsymbol{x})\}$, CVAEs formulate maximization of the conditional log-likelihood $\ln p_\theta(\boldsymbol{x}|\boldsymbol{g})$, parameterized by θ, by considering the following identity:

$$\ln p_\theta(\boldsymbol{x}|\boldsymbol{g}) - D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g}), p_\theta(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})) = \tag{1}$$
$$-D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g}), p(\boldsymbol{z}|\boldsymbol{g})) + \mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})}[\ln p_\theta(\boldsymbol{x}|\boldsymbol{g}, \boldsymbol{z})].$$

Hereby, $D_{\mathrm{KL}}(\cdot, \cdot)$ denotes the Kullback-Leibler (KL) divergence between two distributions, and $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$ is used to approximate the intractable posterior $p_\theta(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$ of a deep net which models the conditional $p_\theta(\boldsymbol{x}|\boldsymbol{g}, \boldsymbol{z})$. The approximation of the posterior, i.e., $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$, is referred to as the encoder, while the deep net used for reconstruction, i.e., for modeling the conditional $p_\theta(\boldsymbol{x}|\boldsymbol{g}, \boldsymbol{z})$, is typically called the decoder.

Since the KL-divergence is non-negative, we obtain a lower bound on the data log-likelihood $\ln p_\theta(\boldsymbol{x}|\boldsymbol{g})$ when considering the right hand side of the identity given in Eq. 1. CVAEs minimize the negated version of this lower bound, i.e.,

$$\min_{\theta, \phi} D_{\mathrm{KL}}(q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g}), p(\boldsymbol{z}|\boldsymbol{g})) - \frac{1}{N}\sum_{i=1}^{N} \ln p_\theta(\boldsymbol{x}|\boldsymbol{g}, \boldsymbol{z}^i), \tag{2}$$

where the expectation $\mathbb{E}_{q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})}$ is approximated via N samples $\boldsymbol{z}^i \sim q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$. For simplicity of the exposition, we ignored the summation over the samples in the dataset $\mathcal{D}$, and provide the objective for training of a single pair $(\boldsymbol{x}, \boldsymbol{g})$.

We next discuss how we combine those ingredients for diverse, controllable yet structurally coherent colorization.
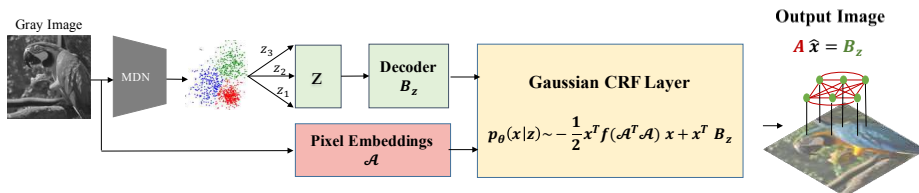
Fig. 2: A fully connected Gaussian Conditional Random Field (G-CRF) based VAE for diverse and globally coherent colorization. To generate diverse colorizations, we use a Mixture Density Network (MDN) to represent the multi-modal distribution of the color field embedding z given the gray-level image g. At test time, we sample multiple embeddings that are subsequently decoded to generate different colorizations. To ensure global consistency, we model the output space distribution of the decoder using a G-CRF.

3 Consistency and Controllability for Colorization

Our proposed colorization model has several appealing properties: (1) *diversity*, i.e., it generates diverse and realistic colorizations for a single gray-level image; (2) *global coherence*, enforced by explicitly modeling the output-space distribution of the generated color field using a fully connected Gaussian Conditional Random field (G-CRF); (3) *controllability*, i.e., our model can consider external constraints at run time efficiently. For example, the user can enforce a given object to have a specific color or two separated regions to have the same colorization.

3.1 Overview

We provide an overview of our approach in Fig. 2. Given a gray-level image g with P pixels, our goal is to produce different color fields $x \in \mathbb{R}^{2P}$ consisting of two channels $x_a \in \mathbb{R}^P$ and $x_b \in \mathbb{R}^P$ in the *Lab* color space. In addition, we enforce spatial coherence at a global scale and enable controllability using a Gaussian Markov random field which models the output space distribution.

To produce a diverse colorization, we want to learn a multi-modal conditional distribution $p(x|g)$ of the color field x given the gray-level image g. However, learning this conditional is challenging since the color field x and the intensity field g are high dimensional. Hence, training samples for learning $p(x|g)$ are sparsely scattered and the distribution is difficult to capture, even when using large datasets. Therefore, we assume the manifold hypothesis to hold, and we choose to learn a conditional $p(x|z, g)$ based on low-dimensional embeddings z captured from x and g, by using a variational auto-encoder which approximates the intractable posterior $p(z|x, g)$ via an encoder. Deshpande *et al.* [15] demonstrated that sampling from the approximation of the posterior results in low variance of the generated images. Following [15], we opt for a multi-stage training procedure to directly sample from $p(z|g)$ as follows.
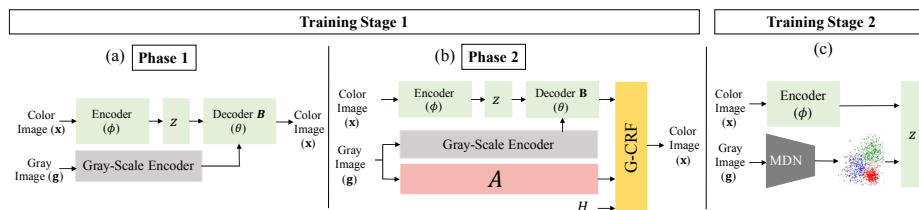
Fig. 3: Overview of the model architecture and the training procedure. In the first training stage, we learn a low dimensional embedding z of the *color field* x conditioned on the gray-level image g using a VAE. To disentangle color from structure, we first learn the unary term B in *phase 1*, then in *phase 2*, learn a precision matrix that encodes the *structure* of the image by imposing the constraint that pixels with similar intensities should have similar colorizations. To enable controllability, we use a training schedule specified in the matrix H to incrementally mask the decoded pixel colors in the unary term B and hence gradually rely on the A matrix to restore the colorization from the unary term. In the second training stage, we use an MDN to learn a multi-modal distribution of the latent embedding given the gray-level image.

To capture the low-dimensional embedding, in a *first training stage*, we use a variational auto-encoder to learn a parametric uni-modal Gaussian encoder distribution $q_\phi(z|x,g) \sim \mathcal{N}(\mu_\phi, \sigma_\phi^2 I)$ of the color field embedding z given both the gray-level image g and the color image x (Fig. 3 (a)). At the same time, we learn the parameters θ of the decoder $p_\theta(x|z,g)$.

Importantly, we note that the encoder $q_\phi(z|x,g)$ takes advantage of both the color image x and the gray-level intensities g when mapping to the latent representation z. Due to the use of the color image, we expect that this mapping can be captured to a reasonable degree using a uni-modal distribution, i.e., we use a Gaussian.

However, multiple colorizations can be obtained from a gray-scale image g during inference. Hence, following Deshpande *et al.* [15], we don't expect a uni-modal distribution $p(z|g)$ to be accurate during testing, when only conditioning on the gray-level image g.

To address this issue, in a *second training stage*, we train a Mixture Density Network (MDN) $p_\psi(z|g)$ to maximize the log-likelihood of embeddings z sampled from $q_\phi(z|x,g)$ (Fig. 3 (b)). Intuitively, for a gray-level image, the MDN predicts the parameters of M Gaussian components each corresponding to a different colorization. The embedding z that was learned in the first stage is then tied to one of these components. The remaining components are optimized by close-by gray-level image embeddings.

At test time, N different embeddings $\{z\}_{k=1}^N$ are sampled from the MDN $p_\psi(z|g)$ and transformed by the decoder into diverse colorizations, as we show in Fig. 2.

To encourage globally coherent colorizations and to ensure controllability, we use a fully connected G-CRF layer to model the output space distribution. The negative log-posterior of the G-CRF has the form of a quadratic energy function:

$$E(\boldsymbol{x}) = \frac{1}{2}\boldsymbol{x}^T \boldsymbol{A_g}\boldsymbol{x} - \boldsymbol{B_{z,g}}\boldsymbol{x}. \tag{3}$$

It captures unary and higher order correlations (HOCs) between the pixels' colors for the a and b channels. Intuitively, the joint G-CRF enables the model to capture more global image statistics which turn out to yield more spatially coherent colorizations as we will show in Sec. 4. The unary term $\boldsymbol{B_{z,g}}$ is obtained from the VAE decoder and encodes the color per pixel. The HOC term $\boldsymbol{A_g} = f(\boldsymbol{\mathcal{A}_g^T}\boldsymbol{\mathcal{A}_g})$ is responsible for encoding the structure of the input image. It is a function of the inner product of low rank pixel embeddings $\boldsymbol{\mathcal{A}_g}$, learned from the gray-level image and measuring the pairwise similarity between the pixels' intensities. The intuition is that pixels with similar intensities should have similar colorizations. The HOC term is shared between the different colorizations obtained at test time. Beyond global consistency, it also enables controllability by propagating user edits encoded in the unary term properly. Due to the symmetry of the HOC term, the quadratic energy function has a unique global minimum that can be obtained by solving the system of linear equations:

$$\boldsymbol{A_g}\boldsymbol{x} = \boldsymbol{B_{z,g}}. \tag{4}$$

Subsequently, we drop the dependency of $\boldsymbol{A}$ and $\boldsymbol{B}$ on $\boldsymbol{g}$ and $\boldsymbol{z}$ for notational simplicity.

We now discuss how to perform inference in our model and how to learn the model parameters such that colorization and structure are disentangled and controllability is enabled by propagating user strokes.

3.2   Inference

In order to ensure a globally consistent colorization, we take advantage of the structure in the image. To this end, we encourage two pixels to have similar colors if their intensities are similar. Thus, we want to minimize the difference between the color field $\boldsymbol{x}$ for the a and b channels and the weighted average of the colors at similar pixels. More formally, we want to encourage the equalities $\boldsymbol{x}_a = \hat{\boldsymbol{S}}\boldsymbol{x}_a$ and $\boldsymbol{x}_b = \hat{\boldsymbol{S}}\boldsymbol{x}_b$, where $\hat{\boldsymbol{S}} = \mathrm{softmax}(\boldsymbol{\mathcal{A}^T}\boldsymbol{\mathcal{A}})$ is a similarity matrix obtained from applying a softmax function to every row of the matrix resulting from $\boldsymbol{\mathcal{A}^T}\boldsymbol{\mathcal{A}}$. To simplify, we use the block-structured matrix $\boldsymbol{S} = \mathrm{diag}(\hat{\boldsymbol{S}}, \hat{\boldsymbol{S}})$.

In addition to capturing the structure, we obtain the color prior and controllability by encoding the user input in the computed unary term $\boldsymbol{B}$. Hence, we add the constraint $\boldsymbol{Hx} = \boldsymbol{\alpha}$, where $\boldsymbol{H}$ is a diagonal matrix with 0 and 1 entries corresponding to whether the pixel's value isn't or is specified by the user, and $\boldsymbol{\alpha}$ a vector encoding the color each pixel should be set to.

With the aforementioned intuition at hand we obtain the quadratic energy function to be minimized as:

$$E_{\theta,\boldsymbol{g},\boldsymbol{z}}(\boldsymbol{x}) = \frac{1}{2}\|(\boldsymbol{I} - \boldsymbol{S})\boldsymbol{x}\|^2 + \frac{1}{2}\beta\|\boldsymbol{Hx} - \boldsymbol{\alpha}\|^2,$$
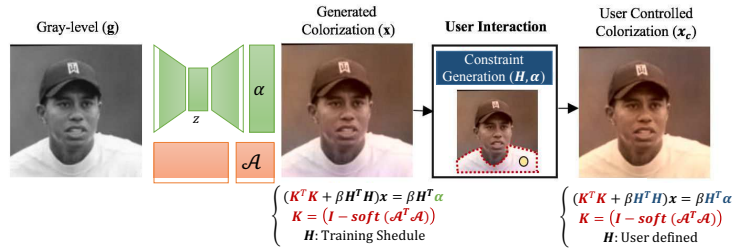
Fig. 4: Controllability: Given a gray-level image, we learn to disentangle structure from colorization. The HOC term is used to propagate sparse user edits encoded in the $\boldsymbol{H}$ and $\boldsymbol{\alpha}$ terms.

with β being a hyper-parameter. This corresponds to a quadratic energy function of the form $\frac{1}{2}\boldsymbol{x}^T\boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{x} + C$, where $\boldsymbol{A} = (\boldsymbol{S} - \boldsymbol{I})^T(\boldsymbol{S} - \boldsymbol{I}) + \beta\boldsymbol{H}^T\boldsymbol{H}$, $\boldsymbol{B} = -2\beta\boldsymbol{\alpha}^T\boldsymbol{H}$ and $C = \beta\boldsymbol{\alpha}^T\boldsymbol{\alpha}$. It's immediately apparent that the unary term only encodes color statistics while the HOC term is only responsible for structural consistency. Intuitively, the conditional $p_\theta(\boldsymbol{x}|\boldsymbol{g}, \boldsymbol{z})$ is interpreted as a Gaussian multi-variate density:

$$p_\theta(\boldsymbol{x}|\boldsymbol{z}, \boldsymbol{g}) \propto \exp(-E_{\theta,\boldsymbol{g},\boldsymbol{z}}(\boldsymbol{x})), \tag{5}$$

parametrized by the above defined energy function $E_{\theta,\boldsymbol{g},\boldsymbol{z}}$. It can be easily checked that $\boldsymbol{A}$ is a positive definite full rank matrix. Hence, for a strictly positive definite matrix, inference is reduced to solving a linear system of equations:

$$((\boldsymbol{I} - \boldsymbol{S})^T(\boldsymbol{I} - \boldsymbol{S}) + \beta\boldsymbol{H}^T\boldsymbol{H})\boldsymbol{x} = \beta\boldsymbol{H}^T\boldsymbol{\alpha}. \tag{6}$$

We solve the linear system above using the LU decomposition of the $\boldsymbol{A}$ matrix. How to learn the terms $\boldsymbol{\alpha}$ and $\boldsymbol{S}$ will be explained in the following.

3.3   Learning

We now present the two training stages illustrated in Fig. 3 to ensure color and structure disentanglement and to produce diverse colorizations. We also discuss the modifications to the loss given in Eq. 2 during each stage.

Stage 1: Training a Structured Output Space Variational Auto-Encoder: During the first training stage, we use the variational auto-encoder formulation to learn a low-dimensional embedding for a given color field. This stage is divided into two phases to ensure color and structure disentanglement. In a first phase, we learn the unary term produced by the VAE decoder. In the second phase, we fix the weights of the VAE apart from the decoder's two top-most layers and learn a D-dimensional embedding matrix $\boldsymbol{\mathcal{A}} \in \mathbb{R}^{D \times P}$ for the P pixels from the gray-level image. The matrix $\hat{\boldsymbol{S}}$ obtained from applying a softmax to every row of $\boldsymbol{\mathcal{A}}^T\boldsymbol{\mathcal{A}}$ is used to encourage a smoothness prior $\boldsymbol{x} = \boldsymbol{S}\boldsymbol{x}$ for the a and b channels. In order to ensure that the $\boldsymbol{S}$ matrix learns the structure required for the

controllability stage, where sparse user edits need to be propagated, we follow a training schedule where the unary terms are masked gradually using the $\boldsymbol{H}$ matrix. The input image is reconstructed from the sparse unary entries using the learned structure. When colorization from sparse user edits is desired, we solve the linear system from Eq. 6 for the learned HOC term and an $\boldsymbol{H}$ matrix and $\boldsymbol{\alpha}$ term encoding the user edits, as illustrated in Fig. 4. We explain the details of the training schedule in the experimental section.

Given the new formulation of the G-CRF posterior, the program for the first training stage reads as follows:

$$\min_{\phi,\theta} D_{\mathrm{KL}}(\mathcal{N}(\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi^2 \boldsymbol{I})), \mathcal{N}(0, \boldsymbol{I})) - \frac{1}{N} \sum_{i=1}^{N} \ln p_\theta(\boldsymbol{x}|\boldsymbol{z}^{(i)}, \boldsymbol{g}) \text{ s.t. } \boldsymbol{z}^{(i)} \sim \mathcal{N}(\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi^2 \boldsymbol{I}). \quad (7)$$

Subsequently we use the term L to refer to the objective function of this program. **Stage 2: Training a Mixture Density Network (MDN):** Since a color image $\boldsymbol{x}$ is not available during testing, in the second training stage, we capture the approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$, a Gaussian which was learned in the first training stage, using a parametric distribution $p_\psi(\boldsymbol{z}|\boldsymbol{g})$. Due to the dependence on the color image $\boldsymbol{x}$ we expect the approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$ to be easier to model than $p_\psi(\boldsymbol{z}|\boldsymbol{g})$. Therefore, we let $p_\psi(\boldsymbol{z}|\boldsymbol{g})$ be a Gaussian Mixture Model (GMM) with M components. Its means, variances, and component weights are parameterized via a mixture density network (MDN) with parameters ψ. Intuitively, for a given gray-level image, we expect the M components to correspond to different colorizations. The colorfield embedding $\boldsymbol{z}$ learned from the first training stage is mapped to one of the components by minimizing the negative conditional log-likelihood, i.e., by minimizing:

$$-\ln p_\psi(\boldsymbol{z}|\boldsymbol{g}) = -\ln \sum_{i=1}^{M} \pi_{\boldsymbol{g},\psi}^{(i)} \mathcal{N}(\boldsymbol{z}|\boldsymbol{\mu}_{g,\psi}^{(i)}, \sigma). \quad (8)$$

Hereby, $\pi_{\boldsymbol{g},\psi}^{(i)}$, $\boldsymbol{\mu}_{\boldsymbol{g},\psi}^{(i)}$ and σ refer to, respectively, the mixture coefficients, the means and a fixed co-variance of the GMM learned by an MDN network parametrized by ψ. However, minimizing $-\ln p_\psi(\boldsymbol{z}|\boldsymbol{g})$ is hard as it involves the computation of the logarithm of a summation over the different exponential components. To avoid this, we explicitly assign the code $\boldsymbol{z}$ to that Gaussian component m, which has its mean closest to $\boldsymbol{z}$, i.e., $m = \underset{i}{\operatorname{argmin}} \|\boldsymbol{z} - \boldsymbol{\mu}_{g,\psi}^{(i)}\|$. Hence, the negative log-likelihood loss $-\ln p_\psi(\boldsymbol{z}|\boldsymbol{g})$ is reduced to solving the following program:

$$\min_{\psi} -\ln \pi_{\boldsymbol{g},\psi}^{(m)} + \frac{\|\boldsymbol{z} - \boldsymbol{\mu}_{\boldsymbol{g},\psi}^{(m)}\|^2}{2\sigma^2} \text{ s.t. } \begin{cases} \boldsymbol{z} \sim q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g}) = \mathcal{N}(\boldsymbol{\mu}_\phi, \boldsymbol{\sigma}_\phi^2 \boldsymbol{I}) \\ m = \underset{i \in \{1,...,M\}}{\operatorname{argmin}} \|\boldsymbol{z} - \boldsymbol{\mu}_{g,\psi}^{(i)}\| \end{cases} . \quad (9)$$

Note that the latent samples $\boldsymbol{z}$ are obtained from the approximate posterior $q_\phi(\boldsymbol{z}|\boldsymbol{x}, \boldsymbol{g})$ learned in the first stage.
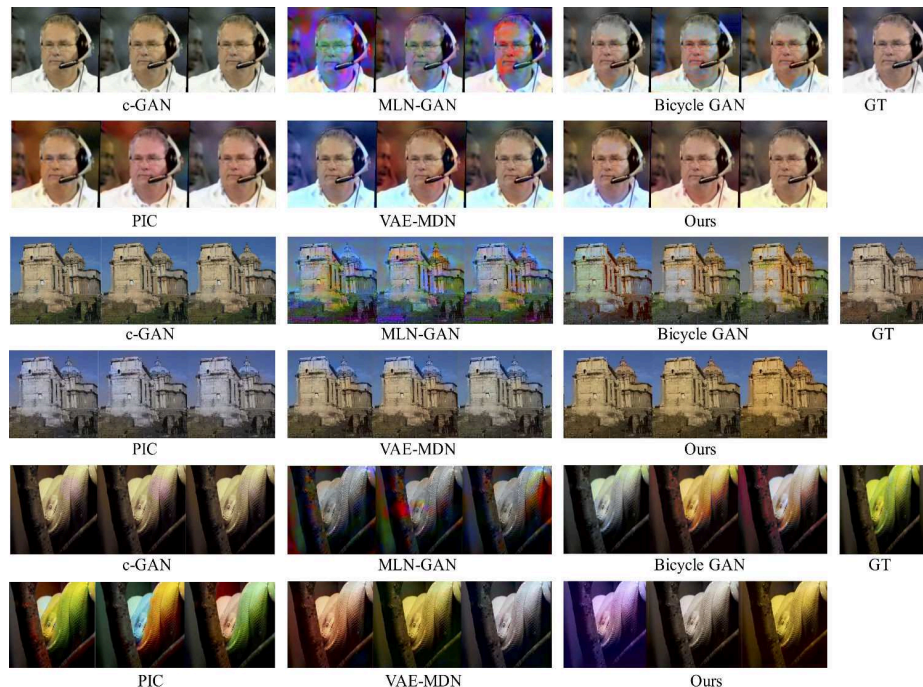
Fig. 5: Qualitative comparison of diverse colorizations obtained from c-GAN [16], MLN-GAN [19], BicycleGAN [20], PIC [18], VAE-MDN [15] and our approach.

4   Results

Next, we present quantitative and qualitative results on three datasets of increasing color field complexity: (1) the Labelled Faces in the Wild dataset (LFW) [22], which consists of 13,233 face images aligned by deep funneling [48]; (2) the LSUN-Church dataset [23] containing 126,227 images and (3) the validation set of ILSVRC-2015 (ImageNet-Val) [24] with 50,000 images. We compare the diverse colorizations obtained by our model with three baselines representing three different generative models: (1) the Conditional Generative Adversarial Network [16, 19, 20]; (2) the Variational Auto-encoder with MDN [15]; and (3) the Probabilistic Image Colorization model [18] based on PixelCNN. Note that [15] presents a comparison between VAE-MDN and a conditional VAE, demonstrating the benefits of the VAE-MDN approach.

4.1   Baselines

Conditional Generative Adversarial Network: We compare our approach with three GAN models: the c-GAN architecture proposed by Isola *et al.* [16], the GAN with multi-layer noise by Cao *et al.* [19] and the BicycleGAN by Zhu *et al.* [20].

Table 1: Results of the user study (% of the model in **bold** winning).

	Ours vs VAE-MDN	**Ours** vs PIC	**VAE-MDN** vs PIC
LFW	61.12 %	59.04 %	57.17 %
LSUN-Church	66.89 %	71.61 %	54.46 %
ILSVRC-2015	54.79 %	66.98 %	62.88%

Table 2: Quantitaive comparison with baselines. We use the error-of-best per pixel (Eob.), the variance (Var.), the mean structural similarity SSIM across all pairs of colorizations generated for one image (SSIM.) and the training time (Train.) as performance metrics.

	LFW				LSUN-Church				ILSVRC-2015			
Method	eob.	Var.	SSIM.	Train.	eob.	Var.	SSIM.	Train.	eob.	Var.	SSIM.	Train.
c-GAN[16]	.047	$8.40e^{-6}$	.92	$\sim$**4**h	.048	$6.20e^{-6}$	.94	$\sim$**39**h	.048	$8.88e^{-6}$	.91	$\sim$**18**h
MLN-GAN[19]	.057	$2.83e^{-2}$	**.12**	$\sim$**4**h	.051	$2.48e^{-2}$	**.34**	$\sim$**39**h	.063	$1.73e^{-2}$	.38	$\sim$**18**h
BicycleGAN[20]	.045	$6.50e^{-3}$	.51	$\sim$**4**h	.048	$2.20e^{-2}$	.38	$\sim$**39**h	.042	$2.20e^{-2}$	**.15**	$\sim$**18**h
VAE-MDN[15]	.035	$1.81e^{-2}$	.49	$\sim$**4**h	.028	**$1.05e^{-2}$**	.77	$\sim$**39**h	.033	$7.17e^{-3}$	.48	$\sim$**18**h
PIC[18]	.043	**$5.32e^{-2}$**	.36	$\sim$48h	.047	$7.40e^{-5}$	.91	$\sim$144h	.035	**$6.74e^{-2}$**	.19	$\sim$96h
Ours	**$11e^{-5}$**	$8.86e^{-3}$	.61	$\sim$**4**h	**$93e^{-6}$**	$1.17e^{-2}$	.83	$\sim$**39**h	**$12e^{-5}$**	$8.80e^{-3}$	.52	$\sim$**18**h

Variational Auto-Encoder with Mixture Density Network (VAE-MDN): The architecture by Deshpande *et al.* [15] trains an MDN based auto-encoder to generate different colorizations. It is the basis for our method.

Probabilistic Image Colorization (PIC): The PIC model proposed by Royer *et al.* [18] uses a CNN network to learn an embedding of a gray-level images, which is then used as input for a PixelCNN network.

Comparison with Baselines: We qualitatively compare the diversity and global spatial consistency of the colorizations obtained by our models with the ones generated by the aforementioned baselines, in Figures 1 and 5. We observe that our approach is the only one which generates a consistent colorization of the skin of the girl in Fig. 1. We are also able to uniformly color the ground, the snake, and the actor's coat in Fig. 5.

For global consistency evaluation, we perform a user study, presented in Tab. 1, where participants are asked to select the more realistic image from a pair of images at a time. We restrict the study to the three approaches with the overall lowest error-of-best (eob) per pixel reported in Tab. 2, namely VAE-MDN, PIC and our model. We use the clicking speed to filter out inattentive participants. Participants did neither know the paper content nor were the methods revealed to them. We gathered 5,374 votes from 271 unique users. The results show that users prefer results obtained with the proposed approach.
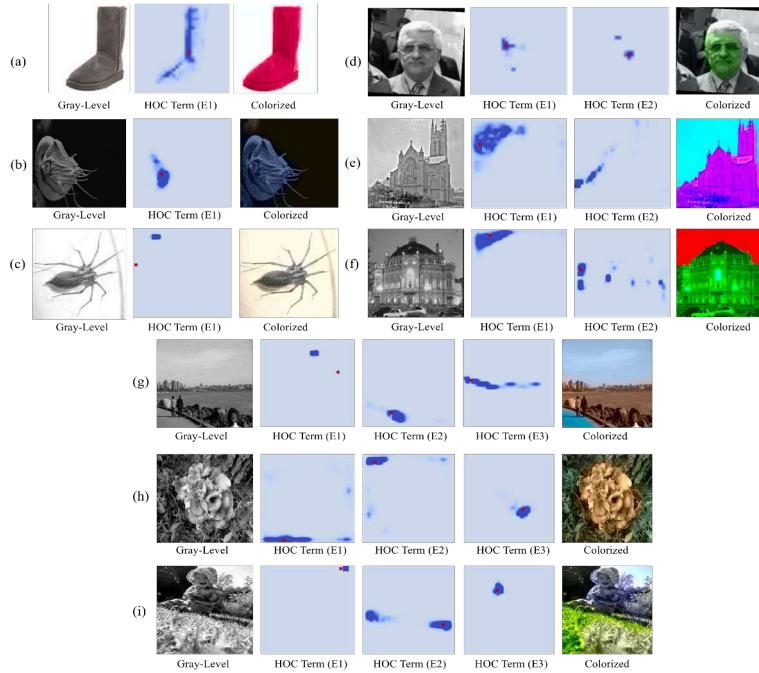
Fig. 6: Controllability: Colorization from sparse user edits.

To evaluate diversity, we use two metrics: (1) the variance of diverse colorizations and (2) the mean structural similarity *SSIM* [49] across all pairs of colorizations generated for one image. We report our results in Tab. 2.

Global Consistency: Our model noticeably outperforms all the baselines in producing spatially coherent results as demonstrated by the user study. PIC generates very diversified samples for the LFW and ILSVRC-2015 datasets but lacks long range spatial dependencies because of the auto-regressive nature of the model. For example, the snake in the second row of Fig. 5 has different colors for the head and the tail, and the woman's skin tone is inconsistent in Fig. 1. The VAE-MDN, BicycleGAN and MLN-GAN outputs are sometimes speckled and objects are not uniformly colored. For example, parts of the dome of the building in the second row of Fig. 5 are confused to be part of the sky and the shirt in the third row is speckled. In contrast, our model is capable of capturing complex long range dependencies. This is confirmed by the user study.

Diversity: Across all datasets, c-GAN suffers from mode collapse and is frequently unable to produce diverse colorizations. The PIC, MLN-GAN and BicycleGAN models yield the most diverse results at the expense of photo-realism. Our model produces diverse results while ensuring long range spatial consistency.

Controllability: For the controllably experiments, we set the β hyper-parameter to 1 during training and to 5 during testing. We opt for the following training schedule, to force the model to encode the structure required to propagate sparse

Table 3: Average PSNR (dB) (higher is better) vs. number of revealed points ($|\boldsymbol{H}|$).

	Levin et al. [2]			Endo et al. [50]			Barron et al. [51]			Zhang et al. [14]			**Ours**		
$|\boldsymbol{H}|$	10	50	100	10	50	100	10	50	100	10	50	100	10	50	100
PSNR	26.5	28.5	30	24.8	25.9	26	25.3	28	29	28	30.2	31.5	26.7	29.3	30.4

Fig. 7: Visualization of the unary term. The first row corresponds to the ground truth image. We visualize one possible colorization in the third row and its corresponding unary term in the second row.

user inputs in the controllability experiments: We train the unary branch for 15 epochs (Stage1, Phase1), then train the HOC term for 15 epochs as well (Stage1, Phase2). We use the diagonal matrix $\boldsymbol{H}$ to randomly specify L pixels which colors are encoded by the unary branch $\boldsymbol{\alpha}$. We decrease L following a training schedule from 100% to 75%, 50%, 25% then 10% of the total number of pixels after respectively epochs 2, 4, 6, 8, 10 and 12. Note that additional stages could be added to the training schedule to accommodate for complex datasets where very sparse user input is desired. In Fig. 6, we show that with a single pixel as a user edit ($E1$), we are able to colorize a boot in pink, a sea coral in blue and the background behind the spider in yellow in respectively Fig. 6 (a-c). With two edits ($E1$ and $E2$), we colorize a face in green (Zhang *et al.* [14] use 3 edits) in Fig. 6 (d) and the sky and the building in different colors in Fig. 6 (e,f). With three user edits ($E1$, $E2$ and $E3$), we show that we can colorize more complex images in Fig. 6 (g-i). We show the edits E using red markers. We visualize the attention weights per pixel, corresponding to the pixel's row in the similarity matrix $\boldsymbol{S}$, in blue, where darker shades correspond to stronger correlations.

Quantitatively, we report the average PSNR for 10, 50 and 100 edits on the ImageNet test set in Tab. 3, where edits (points) corresponding to randomly selected 7×7 patches are revealed to the algorithm. We observe that our method achieves slightly better results than the one proposed by Levin *et al.* [2] as our algorithms learns for every pixel color an 'attention mechanism' over all the pixels in the image while Levin *et al.* impose local smoothness.

Visualization of the HOC and Unary Terms: In order to obtain more insights into the model's dynamics, we visualize the unary terms, $\boldsymbol{B}$, and the HOC terms, $\boldsymbol{A}$, in respectively Fig. 7 and Fig. 8. As illustrated in Fig. 8, the HOC term has learned complex long range pixel affinities through end-to-end
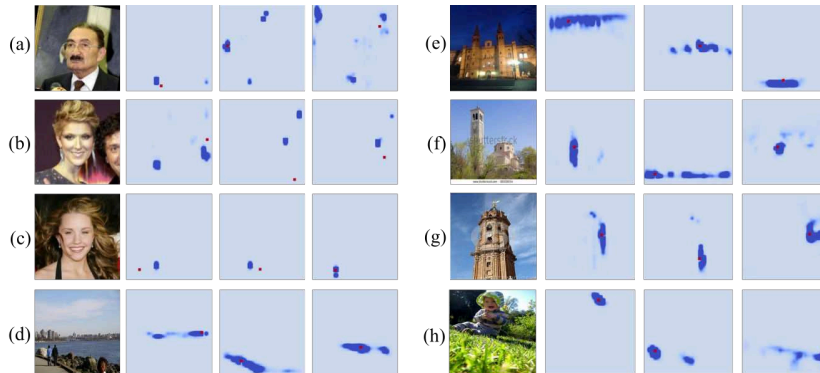
Fig. 8: Visualization of the HOC term. For every example, we show the ground truth image and three HOC terms corresponding to three different pixels marked in red.

training. The results in Fig. 7 further suggest that the unary term outputs a colorization with possibly some noise or inconsistencies that the HOC term fixes to ensure global coherency. For example, for the picture in the second column in Fig. 7, the colors of the face, chest and shoulder predicted by the unary term are not consistent, and were fixed by the binary term which captured the long range correlation as it is shown in Fig. 8 (c).

We notice different interesting strategies for encoding the long range correlations: On the LSUN-Church dataset, the model encourages local smoothness as every pixel seems to be strongly correlated to its neighbors. This is the case for the sky in Fig. 8 (e). The model trained on the LFW dataset, however encoded long range correlation. To ensure consistency over a large area, it chooses some reference pixels and correlates every pixel in the area, as can be seen in Fig. 8 (c).

We provide more results and details of the employed deep net architectures in the supplementary material.

5   Conclusion

We proposed a Gaussian conditional random field based variational auto-encoder formulation for colorization and illustrated its efficacy on a variety of benchmark datasets, outperforming existing methods. The developed approach goes beyond existing methods in that it doesn't only model the ambiguity which is inherent to the colorization task, but also takes into account structural consistency.

References

1. Welsh, T., Ashikhmin, M., Mueller, K.: Transferring color to greyscale images. SIGGRAPH (2002)
2. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. SIGGRAPH (2004)
3. Chia, A.Y.S., Zhuo, S., Gupta, R.K., Tai, Y.W., Cho, S.Y., Tan, P., Lin, S.: Semantic colorization with internet images. SIGGRAPH (2011)
4. Gupta, R.K., Chia, A.Y.S., Rajan, D., Ng, E.S., Zhiyong, H.: Image colorization using similar images. ACM Multimedia (2012)
5. Cohen-Or, D., Lischinski, D.: Colorization by example. Eurographics Symposium on Rendering (2005)
6. Morimoto, Y., Taguchi, Y., Naemura, T.: Automatic colorization of grayscale images using multiple images on the web. SIGGRAPH (2009)
7. Charpiat, G., Hofmann, M., Schölkopf, B.: Automatic image colorization via multimodal predictions. ECCV (2008)
8. Deshpande, A., Rock, J., Forsyth, D.: Learning large-scale automatic image colorization. ICCV (2015)
9. Cheng, Z., Yang, Q., Sheng, B.: Deep colorization. ICCV (2015)
10. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Let there be color!: joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification. SIGGRAPH (2016)
11. Larsson, G., Maire, M., Shakhnarovich, G.: Learning representations for automatic colorization. ECCV (2016)
12. Zhang, R., Isola, P., Efros, A.A.: Colorful image colorization. ECCV (2016)
13. Varga, D., Szirányi, T.: Twin deep convolutional neural network for example-based image colorization. CAIP (2017)
14. Zhang, R., Zhu, J.Y., Isola, P., Geng, X., Lin, A.S., Yu, T., Efros, A.A.: Real-time user-guided image colorization with learned deep priors. SIGGRAPH (2017)
15. Deshpande, A., Lu, J., Yeh, M.C., Forsyth, D.: Learning diverse image colorization. CVPR (2017)
16. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. CVPR (2017)
17. Guadarrama, S., Dahl, R., Bieber, D., Norouzi, M., Shlens, J., Murphy, K.: Pixcolor: Pixel recursive colorization. BMVC (2017)
18. Royer, A., Kolesnikov, A., Lampert, C.H.: Probabilistic image colorization. BMVC (2017)
19. Cao, Y., Zhou, Z., Zhang, W., Yu, Y.: Unsupervised diverse colorization via generative adversarial networks. arXiv preprint arXiv:1702.06674 (2017)
20. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: NIPS. (2017)
21. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. ICLR (2014)
22. Learned-Miller, E., Huang, G.B., RoyChowdhury, A., Li, H., Hua, G.: Labeled faces in the wild: A survey. In: Advances in face detection and facial image analysis. (2016)
23. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop. CoRR, abs/1506.03365 (2015)
24. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. IJCV (2015)

25. Varga, D., Szirányi, T.: Twin deep convolutional neural network for example-based image colorization. ICCAIP (2017)
26. van den Oord, A., Kalchbrenner, N., Espeholt, L., Vinyals, O., Graves, A., et al.: Conditional image generation with pixelcnn decoders. NIPS (2016)
27. Bishop, C.M.: Mixture density networks. Aston University (1994)
28. Laffont, P.Y., Ren, Z., Tao, X., Qian, C., Hays, J.: Transient attributes for high-level understanding and editing of outdoor scenes. SIGGRAPH (2014)
29. Cordts, M., Omran, M., Ramos, S., Rehfeld, T., Enzweiler, M., Benenson, R., Franke, U., Roth, S., Schiele, B.: The cityscapes dataset for semantic urban scene understanding. In: CVPR. (2016)
30. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. CVPR (2017)
31. Yu, A., Grauman, K.: Fine-grained visual comparisons with local learning. In: CVPR. (2014)
32. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: ECCV. (2016)
33. Kindermann, R., Snell, J.L.: Markov random fields and their applications. American Mathematical Society (1980)
34. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Distributed Message Passing for Large Scale Graphical Models. In: Proc. CVPR. (2011)
35. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Globally Convergent Dual MAP LP Relaxation Solvers using Fenchel-Young Margins. In: Proc. NIPS. (2012)
36. Schwing, A.G., Hazan, T., Pollefeys, M., Urtasun, R.: Globally Convergent Parallel MAP LP Relaxation Solver using the Frank-Wolfe Algorithm. In: Proc. ICML. (2014)
37. Meshi, O., Schwing, A.G.: Asynchronous Parallel Coordinate Minimization for MAP Inference. In: Proc. NIPS. (2017)
38. Rue, H.: Gaussian markov random fields: theory and applications. CRC press (2008)
39. Vemulapalli, R., Tuzel, O., Liu, M.Y., Chellapa, R.: Gaussian conditional random field network for semantic segmentation. CVPR (2016)
40. Chandra, S., Usunier, N., Kokkinos, I.: Dense and low-rank gaussian crfs using deep embeddings. ICCV (2017)
41. Chandra, S., Kokkinos, I.: Fast, exact and multi-scale inference for semantic image segmentation with deep gaussian crfs. ECCV (2016)
42. Jancsary, J., Nowozin, S., Sharp, T., Rother, C.: Regression tree fieldsan efficient, non-parametric approach to image labeling problems. CVPR (2012)
43. Tappen, M.F., Liu, C., Adelson, E.H., Freeman, W.T.: Learning gaussian conditional random fields for low-level vision. CVPR (2007)
44. Vemulapalli, R., Tuzel, O., Liu, M.Y.: Deep gaussian conditional random field network: A model-based deep network for discriminative denoising. CVPR (2016)
45. Sohn, K., Lee, H., Yan, X.: Learning structured output representation using deep conditional generative models. NIPS (2015)
46. Wang, L., Schwing, A.G., Lazebnik, S.: Diverse and Accurate Image Description Using a Variational Auto-Encoder with an Additive Gaussian Encoding Space. In: NIPS. (2017)
47. Jain*, U., Zhang*, Z., Schwing, A.G.: Creativity: Generating Diverse Questions using Variational Autoencoders. In: Proc. CVPR. (2017) * equal contribution.
48. Huang, G., Mattar, M., Lee, H., Learned-Miller, E.G.: Learning to align from scratch. NIPS (2012)

49. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE TIP (2004)
50. Endo, Y., Iizuka, S., Kanamori, Y., Mitani, J.: Deepprop: Extracting deep features from a single image for edit propagation. Eurographics (2016)
51. Barron, J.T., Poole, B.: The fast bilateral solver. In: ECCV. (2016)