Efficient Learning-based Image Enhancement: Application to Super-resolution and Compression Artifact Removal

Younghee Kwon[1]

Kwang In Kim[2]
http://www.mpi-inf.mpg.de/~kkim

Jin Hyung Kim[3]
http://ai.kaist.ac.kr/~jkim/

Christian Theobalt[2]
http://www.mpi-inf.mpg.de/~theobalt

[1] Google Inc.
1600 Amphitheatre Parkway,
Mountain View, CA, USA

[2] Max-Planck-Institut für Informatik
Campus E1-4,
Saarbrücken, Germany

[3] KAIST
291 Daehak-ro Yuseong-gu,
Daejeon, Korea

Abstract

In this paper, we describe a framework for learning-based image enhancement. At the core of our algorithm lies a generic regularization framework that comprises a prior on natural images, as well as an application-specific conditional model based on Gaussian processes. In contrast to prior learning-based approaches, our algorithm can instantly learn task-specific degradation models from sample images which enables users to easily adopt the algorithm to a specific problem and data set of interest. This is facilitated by our efficient approximation scheme of large-scale Gaussian processes. We demonstrate the efficiency and effectiveness of our approach by applying it to two example enhancement applications: single-image super-resolution as well as artifact removal in JPEG-encoded images.

1 Introduction

Many widely used imaging operations lead to specific degradations of the original image. The removal of these degradations is one of the most important tasks in computer vision, image processing, and computational photography. For instance, image encoding deficiencies such as block artifacts have to be removed frequently. Deterioration and information loss due to the limitations of the optical system, such as limited sensor resolution or defocusing, should also be erased. This paper presents a framework to solve a variety of such enhancement operations, which is based on efficient machine learning of application-specific enhancement models.

Motivated by the recent success of Bayesian approaches in related image enhancement applications [4, 7, 13], we model the imaging process as a combination of a *generic* prior on

natural images and *application-specific* conditional models. We adopt the *product of edge-perts* (PoEdges) model [4] as a prior. This model adopts a sparsity prior (i.e., *Laplacian*) over the pair-wise joint distribution of wavelet coefficients which, overall, prefers simultaneous activation of few coefficients in nearby frequencies, scales, and spatial locations. This enables very efficient modeling of high-order dependencies. The PoEdges model has been successfully applied to denoising under Gaussian noise. However, it is generic and, as we show in this paper, can also handle non-Gaussian noise.

For exploiting the knowledge of a degradation process, we apply non-parametric Gaussian process (GP) regression, instead of the commonly used parametric noise models (Sec. 2). This relieves the user from the extremely difficult task of designing an analytical noise model, in particular for general non-Gaussian noise. Instead, with our framework, one can build an image enhancement system by preparing a set of example pairs of clean and degraded images and learning case specific conditional models from such training data. We are not the first to apply learning-based image enhancement (see the related work paragraph). However, we explicitly overcome the obstructive high run-time complexity of previous approaches. One of our main contributions is that instead of time-consuming training and testing of a single GP model on a large dataset, a set of *sparse* models is constructed *on-line* such that the prediction at each test data point is made by the corresponding sparse GP approximating the underlying global model. We will demonstrate that during inference, i.e. enhancement, this method has a similar run-time complexity and performance as general sparse models [5, 13, 16, 20]. However, unlike existing models, our approach avoids the time-consuming training stage, and therefore facilitates easy customization of the framework to specific image enhancement problems.

Our framework is generic, and is not restricted to any specific application. In the current paper, we demonstrate this with two exemplary image enhancement operations that can benefit from the addition of magnitude higher efficiency: single-image super-resolution and enhancement of JPEG-encoded images (Sec. 3). The experiments demonstrate that the proposed framework is on par with or outperforms state-of-the-art systems in terms of achieved quality and run-time complexity.

Related work. A variety of image enhancement operations can be approached with our framework. In this section, we illustrate this by reviewing related literature from the two specific applications that we exemplify in this paper.

Block-based discrete cosine transform (BDCT) coding is one of the most widely used tools for compressing still images (*e.g.*, JPEG) and video sequences (*e.g.*, MPEG). At low bit rates, BDCT-encoded images can exhibit discontinuities at block boundaries, known as *block artifacts*. Removal of block artifacts is not only an interesting problem by itself but also provides a good example of image enhancement under non-Gaussian noise. Several papers propose adaptive filtering for block artifact removal, i.e. locally adjusting filter kernels to remove block edges while preserving image edges [19]. A similar technique has also been applied to the removal of ringing artifacts in the context of trilateral filters [21]. Zhai *et al.* [24] proposed a block-shift filtering-based algorithm. For each pixel, the algorithm reconstructs a block encompassing that pixel based on a weighted combination of neighboring similar blocks. The overall result is a detail-preserving smoothing.

In general image enhancement, incorporation of *a priori* knowledge about natural images can prove to be very beneficial. In principle, an image model incorporating a generic prior of natural images can be applied to any type of enhancement applications with suitable

modification of the noise model (or even with a Gaussian noise model). The theory of *projection onto convex sets* (POCS) models prior knowledge as a set of convex constraints (*e.g.*, spatial smoothness, quantization constraints) by which the image enhancement is cast into the iteration through POCS. POCS has been successfully applied to JPEG image enhancement [23].

A rather direct way of utilizing *a priori* knowledge is to encode it into a distribution or an energy functional. Sun and Cham [18] proposed a *maximum a posteriori* (MAP) framework where the prior is modeled as a Markov random field with learned potentials. Nosratinia [11] proposed another promising method called *re-application of JPEG*, which repeatedly reapplies JPEG coding to shifted versions of the JPEG coded image, and averages the result.

Machine learning-based approaches for image enhancement learn a mapping from the noise-affected image space to the ground truth image space. Qiu [12] used a multi-layer Perceptron while Lee *et al.* [9] proposed performing the piecewise linear regression in the space of DCT coefficients and showed comparable results to those of re-application of JPEG. Laparra *et al.* [7] recently proposed a generic wavelet domain framework where the distributions of the source (clean image) and the noise were estimated non-parametrically based on support vector regression (SVR).

Single-image super-resolution is the task of constructing a high-resolution enlargement of a single low-resolution image. For this scenario, the most closely related approaches to our proposed framework are the example-based methods of Freeman *et al.* [3] and of Kim and Kwon [5]. In Freeman *et al.*'s algorithm, the stored examples are retrieved through a nearest neighbor (NN)-search that enforces spatial consistency. Kim and Kwon generalized this idea by replacing NN-search with sparse kernel ridge regression and adopting a prior on major edges. In theory, the latter approach could also be generalized to other image enhancement approaches. However, our proposed framework achieves comparable performance with two orders of magnitude faster training time, which dramatically enhances its applicability.

2 The general image enhancement framework

Prior on natural images. For the case of Gaussian noise, the PoEdges model provides a MAP framework in the decorrelated[1] wavelet domain:

$$\mathbf{z}^* \quad = \quad \arg\max_{\mathbf{z}} \left(\log p(\tilde{\mathbf{z}}|\mathbf{z}) + \log p(\mathbf{z})\right) = \arg\min_{\mathbf{z}} \left(\frac{1}{2}\|\tilde{\mathbf{z}} - \mathbf{z}\|^2 + \sigma_P \left(\sum_j w_j [\mathbf{z}]_j^2\right)^{\alpha_P} \right), \quad (1)$$

where $\tilde{\mathbf{z}} = \mathcal{W}[\tilde{\mathbf{x}}]$, $\mathcal{W}[\cdot]$ is the wavelet transform, $\tilde{\mathbf{x}}$ represents the noisy input image, and σ_P is the regularization hyper-parameter which is specified by the user. The parameters of experts model $\{w_j\}$ and $\alpha_P \in [0,1]$ are estimated by an *expectation maximization* type algorithm [9].

A straightforward approach to apply the PoEdges framework to general image enhancement problems is to modify the noise model accordingly:

$$p(\tilde{\mathbf{z}}|\mathbf{z}) \quad \propto \quad \left\| \tilde{\mathbf{z}} - \mathcal{W}[\mathcal{I}[\mathcal{W}^{\#}(\mathbf{z})]] \right\|^2, \quad (2)$$

where $\mathcal{I}[\cdot]$ is the degradation process of interest and $\mathcal{W}^{\#}(\mathbf{z})$ is the preimage of $\mathbf{z}$. However, in general, $\mathcal{I}$ is non-differentiable and it is not even continuous which can lead to difficulties in the optimization.

[1] For brevity, we omit the details of the decorrelation and the estimation procedure for the clusters (see [9]).

To keep computations tractable, we bypass the optimization through the degradation process and penalize instead the cost functional

$$\mathcal{E}(\mathbf{z}) = \frac{1}{2}\|\mathbf{z} - \mathcal{W}[\mathbf{s}]\|^2 + \sigma_P(\sum_j w_j [\mathbf{z}]_j^2)^{\alpha_P}, \tag{3}$$

where the *reference variable* $[\mathbf{s}]_i$ is constructed as a convex combination of candidates $\mathbf{f}_i$ based on their *confidences* $\mathbf{c}_i$ (i.e., $[\mathbf{s}]_i = \mathbf{f}_i^\top \mathbf{c}_i, [\mathbf{c}_i]_j \geq 0, \|\mathbf{c}_i\|_{L_1} = 1$). The confidence vector $\mathbf{c}_i \in \mathbb{R}^N$ is calculated based on the *predictive variances* $\mathbf{v}_i$ of the candidates $\mathbf{f}_i$:

$$[\mathbf{c}_i]_j = \exp\left(-\frac{[\mathbf{v}_i]_j}{\sigma_C}\right) \Big/ \sum_{k=1,\ldots,N} \exp\left(-\frac{[\mathbf{v}_i]_k}{\sigma_C}\right), \tag{4}$$

where the scale parameter σ_C is fixed at 0.2. Both $\mathbf{f}_i \in \mathbb{R}^N$ and $\mathbf{v}_i \in \mathbb{R}^N$ are estimated during the regression step (see the regression paragraph).

In this model, the degradation process is taken into account only indirectly by $\mathbf{s}$ which encodes the information contained in the training examples. The model is computationally favorable. Unfortunately, in this case an intuitive probabilistic interpretation of the image enhancement process is not possible. Especially, since the conditional model is not generative, sampling a degraded image is not directly possible. However, for image enhancement applications, this is not a serious problem. A similar strategy has been exercised in [2].

Regression. For each pixel location in the input degraded image $\tilde{\mathbf{x}}$, a Gaussian process (GP) regressor receives a patch (of size $M \times M$) centered at that location, and produces estimates of a desired patch (of size $N \times N$). GP prediction is obtained as a Gaussian distribution which in the current context of patch regression, is specified by the mean patch and the corresponding variance patch. When $N > 1$, the output patch overlaps its spatial neighbors, which for each pixel location i, constitutes a set of candidates for the pixel values $\mathbf{f}_i$ and the corresponding predictive variances $\mathbf{v}_i$, respectively.

Suppose there is a given set of data points (i.e. patches) $\mathcal{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_l\} \subset \mathbb{R}^{M^2}$ and their corresponding labels $\mathcal{Y} = \{\mathbf{y}_1, \ldots, \mathbf{y}_l\} \subset \mathbb{R}^{N^2}$. We adopt a Gaussian noise model with mean $\mathbf{0}$ and the covariance matrix $\sigma^2\mathbf{I}$

$$\mathbf{y}_i = f(\mathbf{x}_i) + \varepsilon, \text{ where } \varepsilon \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}), \tag{5}$$

where $\mathcal{N}(\mu, \Sigma)$ is the probability density of the Gaussian random variable with mean μ and covariance Σ and $f : \mathbb{R}^{M^2} \mapsto \mathbb{R}^{N^2}$ is the underlying *latent* function. Then, a zero-mean Gaussian process (GP) prior is placed over f, which for a given set of test points $\mathcal{X}_* = \{\mathbf{x}_{*(1)}, \ldots, \mathbf{x}_{*(l')}\}$ is realized as [14][2]

$$p(\mathbf{f}_*, \mathbf{f}) = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{K}_{\mathbf{f},\mathbf{f}} & \mathbf{K}_{\mathbf{f},*} \\ \mathbf{K}_{*,\mathbf{f}} & \mathbf{K}_{*,*} \end{bmatrix}\right), \tag{6}$$

where the subscripts $\mathbf{f}$ and $*$ represent indexing across training and testing data points, respectively (*e.g.*, $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_l)]^\top$, $\mathbf{f}_* = [f(\mathbf{x}_{*(1)}), \ldots, f(\mathbf{x}_{*(l')})]^\top$, and $[(\mathbf{K}_{*,\mathbf{f}})_{(i,j)}]_{l',l} =$

[2]Each output is treated independently. For notational convenience, we omit conditioning on input variables.

$k(\mathbf{x}_{*(i)}, \mathbf{x}_j)$). While any positive definite function can be used as the *covariance* function k, we adopt the standard Gaussian kernel

$$k(\mathbf{x}, \mathbf{y}) = \exp\left(-b\|\mathbf{x}-\mathbf{y}\|^2\right). \tag{7}$$

With this prior, the prediction is made by conditioning on the labels, which leads to a *predictive distribution* $p(\mathbf{f}_*|\mathcal{Y})$.

While GP regression has shown to be competitive on a wide range of small-scale applications, its application to large-scale problems is limited due to its unfavorable scaling behavior: The computation of the predictive distribution takes $O(Ml^2 + l^3)$ time off-line plus $O(Ml + lN + l^2)$ per test point (cf. [14] for details).

A standard approximate approach to overcome the unfavorable scaling behavior of GPs is to introduce a small set of *inducing variables* $\mathbf{f}_\mathcal{U} = \{f(\mathbf{u}_1), \ldots, f(\mathbf{u}_m)\}$ (corresponding to *inducing inputs* $\mathcal{U} = \{\mathbf{u}_1, \ldots, \mathbf{u}_m\}$) through which the conditional independence of $\mathbf{f}_*$ and $\mathbf{f}$ is assumed (cf. the unified framework of [13]):

$$p(\mathbf{f}_*, \mathbf{f}) \approx q(\mathbf{f}_*, \mathbf{f}) = \int q(\mathbf{f}_*|\mathbf{f}_\mathcal{U}) q(\mathbf{f}|\mathbf{f}_\mathcal{U}) p(\mathbf{f}_\mathcal{U}) d\mathbf{f}_\mathcal{U}. \tag{8}$$

This leads to a set of approximations which are referred to as *sparse* GPs where the inference is carried out through $\mathcal{U}$ summarizing the entire training set $\mathcal{X}$ [15, 16, 20]. For instance, Seeger *et al.* [15] proposed an approximate prior

$$q(\mathbf{f}_*, \mathbf{f}) = \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \mathbf{Q}_{\mathbf{f},\mathbf{f}} & \mathbf{Q}_{\mathbf{f},*} \\ \mathbf{Q}_{*,\mathbf{f}} & \mathbf{K}_{*,*} \end{bmatrix}\right), \tag{9}$$

where $\mathbf{Q}_{\mathbf{a},\mathbf{b}} = \mathbf{K}_{\mathbf{a},\mathbf{u}}\mathbf{K}_{\mathbf{u},\mathbf{u}}^{-1}\mathbf{K}_{\mathbf{u},\mathbf{b}}$. The corresponding predictive distribution is

$$q(\mathbf{f}_*|\mathcal{Y}) \quad = \quad \mathcal{N}\left(\mathbf{Q}_{*,\mathbf{f}}(\mathbf{Q}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I})^{-1}\mathbf{Y}, \mathbf{K}_{*,*} - \mathbf{Q}_{*,\mathbf{f}}(\mathbf{Q}_{\mathbf{f},\mathbf{f}} + \sigma^2\mathbf{I})^{-1}\mathbf{Q}_{\mathbf{f},*}\right), \tag{10}$$

where $\mathbf{Y}$ is a matrix in which each row corresponds to an element of $\mathcal{Y}$. With this prior, the predictive mean is obtained as a linear combination of evaluations of m basis functions $\{k(\mathbf{u}_1, \cdot), \ldots, k(\mathbf{u}_m, \cdot)\}$ (explaining the name *sparse GPs*). The time complexity of calculating the predictive distribution becomes $O(Mlm + lm^2)$ off-line plus $O(Mm + mN + m^2)$ per test point. The performance of a sparse approximation depends heavily on the inducing inputs $\mathcal{U}$. Several different criteria have been proposed for building $\mathcal{U}$. However, usually they are non-convex and accordingly are not easy to optimize.

In this work, instead of solving a difficult optimization problem, an efficient and effective heuristic is adopted for quickly identifying $\mathcal{U}$: for a given test point $\mathbf{x}_*$, we construct online, a sparse GP model where the nearest neighbors (NNs) $\mathcal{C}_m(\mathbf{x}_*) \subset \mathcal{X}$ of $\mathbf{x}_*$ are used as the inducing inputs $\mathcal{U}_*$. This corresponds to imposing the spatial Markov assumption upon $\{f_*, \mathbf{f}\}$ in the approximation (8).

The spatial Markov model is fairly natural and has proven to be effective in many different applications. Although the resulting sparse model can represent only local variations at $\mathbf{x}_*$, the corresponding prediction takes into account the entire data set through (8) (i.e., $q(f_*, \mathbf{f})$ still fits into the approximation (9)). Accordingly, it is a valid approximation of the full GP. This is in contrast to well-known moving least-squares algorithm which is not directly related to any global regularization.

Table 1: Parameters used in the experiments

Expr. idx.	M	N	σ_P	σ^2	b
JPEG	7	5	2.0	$5*10^{-3}$	10
Super-resolution	7	5	0.5	$5*10^{-8}$	20

This new approximation dramatically reduces the computation time during training. Actually, the only training component is building a structure for NN-search. During the prediction, it takes $O(Mlm+lm^2)$ time for each test point, which includes the time spent for building a model. For large l ($\approx 2*10^5$ in the current applications), this might be still impractical. The second step of approximation is then selecting n NNs $\mathcal{C}_n(\mathbf{x}_*)(\mathcal{C}_m(\mathbf{x}_*) \subset \mathcal{C}_n(\mathbf{x}_*) \subset \mathcal{X})$ of $\mathbf{x}_*$ such that the prediction is performed based on n data points which are summarized by m inducing inputs:

$$q(f_*|\mathcal{Y}(\mathcal{C}_n(\mathbf{x}_*)),\mathbf{x}_*,\mathcal{C}_n(\mathbf{x}_*)) = \mathcal{N}\left(\mathbf{Q}_{*,\mathbf{c}}\Gamma\mathbf{Y}(\mathcal{C}_n(\mathbf{x}_*)),\mathbf{K}_{*,*}-\mathbf{Q}_{*,\mathbf{c}}\Gamma\mathbf{Q}_{\mathbf{c},*}\right), \tag{11}$$

where $\mathcal{Y}(A)(\mathbf{Y}(A))$ represents the subset of $\mathcal{Y}$ (rows of $\mathbf{Y}$) corresponding to the elements of $A \subset \mathcal{X}$, $\Gamma = \left(\mathbf{Q}_{\mathbf{c},\mathbf{c}} + \mathrm{diag}[\mathbf{K}_{\mathbf{c},\mathbf{c}} - \mathbf{Q}_{\mathbf{c},\mathbf{c}} + \sigma^2\mathbf{I}]\right)^{-1}$, and $\mathbf{c}$ represents indexing across $\mathcal{C}_n(\mathbf{x}_*)$.

This step is motivated by the fact that for large l, the predictive mean of the full GP can be well-approximated by a *kernel smoother* [17]:

$$[\mathbb{E}[f(\mathbf{x}_*)]]_j = \sum_{i=1,\ldots,l} \tilde{\kappa}(\|\mathbf{x}_* - \mathbf{x}_i\|)[\mathbf{Y}]_{i,j}, \tag{12}$$

where the *equivalent kernel* $\tilde{\kappa}$ corresponding to k has the localization property (i.e., $\tilde{\kappa}(\|\mathbf{x}_* - \cdot\|)$ diminishes quickly with distance from $\mathbf{x}_*$; see [6] for details). Since the variances of elements of $\mathbf{Y}$ are bounded, the locality of $\tilde{\kappa}$ indicates that the *weight functions* $\{\tilde{\kappa}(\|\mathbf{x}_i - \cdot\|)\}$ corresponding to data points $\mathbf{x}_i$ that are distinct from $\mathbf{x}_*$ do not contribute significantly to the expansion (12). This choice also maximizes the *differential entropy score* [8]. To guarantee that the resulting GPs are non-locally regularized, we set $\mathcal{C}_m$ much wider than $\mathcal{C}_n$.

This new approximation will henceforth be referred to as a *semi-local GP* approximation. An important advantage of the semi-local GP over existing sparse applications is that given hyper-parameters, the only training component is building a data structure for NN-search, making the off-line processing is very fast. Therefore, the framework is very flexible as the system can be easily adapted to the distribution of a specific class of images (see Sec. 3).

3 Results and Discussion

This section demonstrates two different applications of the proposed framework: enhancement of JPEG images and single-image super-resolution. To evaluate the performance in each case, we used 16 images (of size 512×512 or 256×256) shown in Fig. 1, which are disjoint from the training images where 200,000 training data points are sampled for GP regressions. To facilitate quantitative evaluation, for each clean image, a degraded image was generated according to the corresponding task (*e.g.*, JPEG encoding for the enhancement of JPEG images and blurring plus sub-sampling for super-resolution, respectively). The subsequently enhanced images were then compared with the original images. On average, (in *on-line scenario*; see the next paragraph for details) processing a single 512×512-sized image took around two minutes plus five minutes of training (degrading images, sampling training data, and building an NN-search tree) which was done once for all 16 test images

Figure 1: Gallery of test images: the images are referred to in the text by their positions in raster order. The last four images are contained in the Berkeley segmentation dataset [10].

for each task. Table 1 summarizes the parameters used in the experiments. Details of the parameter selection strategy are discussed in the supplementary material.

Enhancement of JPEG-encoded images. We propose two application scenarios for enhancement of compressed images. In the *off-line scenario*, a model specialized to each small interval of compression ratios is trained such that the whole range of compression ratios is covered by several models. Then, the enhancement of a given encoded image can be performed by choosing the closest model based on the compression ratio. In the *on-line scenario*, for every given image a model is instantly trained from the example image pairs generated with the compression ratio at hand.[3] While the proposed framework can afford both application scenarios, in general, the second scenario is preferable since the resulting system is more flexible: the system is specifically tailored for each input image. This becomes feasible due to our semi-local approximation. Basic GP regression for 200,000 data points is entirely infeasible, whereas training the sparse Gaussian process model [5] also took around a day and a half in our preliminary experiments.

In the experiments, we will be focusing on the on-line scenario and on specific compression ratios: the quantization table (which determines the compression ratio) Q2 in Table 2 of [11]. Application examples to other compression ratios are provided in the accompanying supplementary material.

As a preprocessing step, the input JPEG images are first preprocessed by re-application of JPEG [11] which already suppresses block-artifacts in an efficient way, but tends to leave ringing artifacts. Then, similarly to [9], the preprocessed images are band-frequency filtered based on the Laplacian of Gaussian (LOG) filter. Given a patch of LOG filtered image, the regressor estimates a patch corresponding to the difference between the input and the underlying ground truth such that the output candidates $\mathbf{y}$ are obtained by adding the regression result to the input image.

The compression ratios used in the current paper and in the supplementary material, as well as our test set of images (Fig. 1) are also employed in many published JPEG image enhancement works (*e.g.*, [9, 18, 23, 24]), which allows us to compare our method to these approaches. Especially the enhancement results corresponding to the classical standard images (*e.g.*, 'Goldhill', 'Lena', and 'pepper' images: from eighth to tenth images in Fig. 1) reported in these publications indicate that the proposed JPEG image enhancement method is significantly better in terms of PSNR. To compare against most closely related state-of-

[3] In this case, the time consuming parameter optimization can be avoided by optimizing them off-line in a manner similar to the off-line scenario.

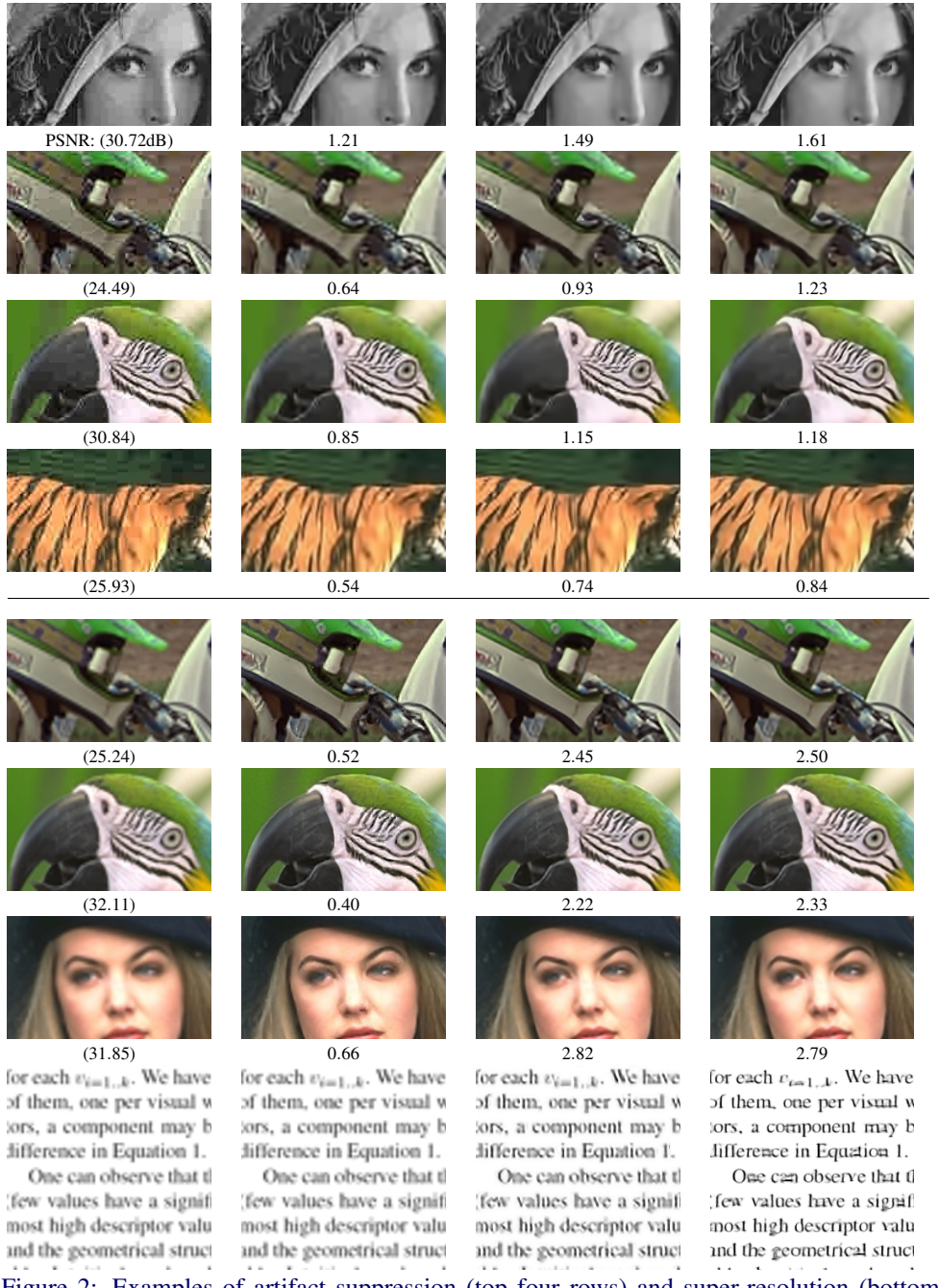

Figure 2: Examples of artifact suppression (top four rows) and super-resolution (bottom four rows). Artifact suppression: from left to right, input JPEG images, re-application of JPEG [11], SADCT [1], and the proposed method. Super-resolution: from left to right, bicubic resampling, Freeman *et al.* [3], Kim and Kwon [5], and the proposed method. Increases of PSNRs with respect to the input degraded images (displayed below each column) were calculated based on the complete images. There is no ground truth for the document image.

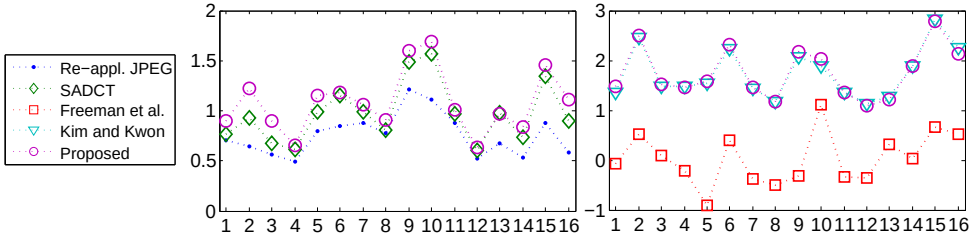

Figure 3: Performance of different JPEG enhancement (left) and super-resolution (right) algorithms: increases of PSNRs (in dB) from JPEG images and bicubic-resampled images, respectively. The x axis corresponds to the image index.

the-art JPEG enhancement approaches, the re-application of JPEG [11] and shape-adaptive DCT [1] are evaluated using publicly available source code.[4] A comparison with another state-of-the-art algorithm proposed by Laparra *et al.* [7] is provided in the accompanying supplementary material, which also shows that our approach performs better. It should be noted that all three methods already outperformed many existing algorithms (cf., comparison with the other algorithms reported in [1, 7, 11]).

Visual inspection of our result to the results obtained with the other methods (Fig. 2), reveals that our method produces the least number of artifacts and preserves actual image features best. This is also numerically confirmed through the PSNR values, Fig. 3.

Single-image super-resolution. Adopting the framework of [3, 5], the input low-resolution image is first enlarged to the target scale by bicubic resampling. Then, we use the same pre-processing steps as for JPEG enhancement, and apply our framework. We will focus on a magnification by 2 along each dimension. Results for other factors are provided in the supplementary material.

Figures 2 and 3 show the results of super-resolution. For comparison, the results of Freeman *et al.*'s algorithm [3] and Kim and Kwon's algorithm [5] are also displayed. A comparison with another recent learning-based algorithm [22] is provided in the supplementary material.

All tested super-resolution algorithms outperformed the simple baseline method, namely, bicubic resampling. Freeman *et al.*'s algorithm produced sharper but partially noisy images. For general images, the results of our approach and [5] are equally good (Figs. 2 and 3). They are as sharp as [3] but exhibit a lot less noise. It should be noted that Kim and Kwon's algorithm already outperformed several state-of-the-art algorithms. However, for a given special class of images whose statistical properties might be distinct from those of general images (*e.g.*, documents as exemplified in the last row of Fig. 2), the proposed method can produce much better results, since in this framework, one can instantly generate a model specific to the given class of images, which is infeasible in [5] due to its high complexity in training. For the document image, the proposed method was trained based on a document image DB (with different sizes, fonts, resolutions, etc.). The hyper-parameters for our document-specific system were adopted from the same system trained on the generic DB so that the time-consuming parameter optimization stage was avoided.

[4] http://www.cs.tut.fi/~foi/SA-DCT/

Discussion. Our framework can be quickly applied to new problems, even by users with no specific knowledge of the image enhancement operation to be performed. In this paper, we have focused on designing a general framework and have not tried to maximize the contribution of application-specific components. Although the resulting systems already demonstrated state-of-the-art performance or even better than state-of-the-art performance in example restoration tasks, we expect that the enhancement quality could be even further improved by resorting to more advanced image representations or preprocessing. Future work should explore this direction.

Nonetheless, our framework provides interesting conceptual insights, allows for high-quality image enhancement in different scenarios, and allows us to customize the degradation models very efficiently since training times are orders of magnitude faster than related previous methods from the literature.

Acknowledgment

The ideas presented in this study have greatly profited from discussions with Christian Walder, James Tompkin, Nils Hasler, and Miguel Granados. Part of this work was done while Y. Kwon was with KAIST and K. I. Kim was with machine learning group, Saarland University.

References

[1] A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise shape-adaptive DCT for high-quality denoising and deblocking of grayscale and color images. *IEEE Trans. Image Processing*, 16(5):1395–1411, 2007.

[2] W. T. Freeman, E. C. Pasztor, and O. T. Carmichael. Learning low-level vision. *International Journal of Computer Vision*, 40(1):25–47, 2000.

[3] W. T. Freeman, T. R. Jones, and E. C. Pasztor. Example-based super-resolution. *IEEE Computer Graphics and Applications*, 22(2):56–65, 2002.

[4] P. V. Gehler and M. Welling. Product of "edge-perts". In *Advances in Neural Information Processing Systems*, Cambridge, MA, 2005. MIT Press.

[5] K. I. Kim and Y. Kwon. Single-image super-resolution using sparse regression and natural image prior. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 32(6): 1127–1133, 2010.

[6] K. I. Kim, Y. Kwon, J.-H. Kim, and C. Theobalt. Efficient learning-based image enhancement: application to compression artifact removal and super-resolution. Technical Report MPI-I-2011-4-002, Max-Planck-Insitut für Informatik, February 2011. URL http://www.mpi-inf.mpg.de/~kkim/imglearn/tr.pdf.

[7] V. Laparra, J. Gutiérrez, G. Camps-Valls, and J. Malo. Image denoising with kernels based on natural image relations. *Journal of Machine Learning Research*, 11:873–903, 2010.

[8] N. Lawrence, M. Seeger, and R. Herbrich. Fast sparse Gaussian process methods: the informative vector machine. In *Advances in Neural Information Processing Systems*, pages 625–632, Cambridge, MA, 2003. MIT Press.

[9] K. Lee, D. S. Kim, and T. Kim. Regression-based prediction for blocking artifact reduction in JPEG-compressed images. *IEEE Trans. Image Processing*, 14(1):36–49, 2005.

[10] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. International Conference on Computer Vision*, volume 2, pages 416–423, 2001.

[11] A. Nosratinia. Denoising of JPEG images by re-application of JPEG. *Journal of VLSI Signal Processing*, 27(1):69–79, 2001.

[12] G. Qiu. MLP for adaptive postprocessing block-coded images. *IEEE Trans. Circuits and Systems for Video Technology*, 10(8):1450–1454, 2000.

[13] J. Quińonero-Candela and C. E. Rasmussen. A unifying view of sparse approximate Gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005. ISSN 1532-4435.

[14] C. E. Rasmussen and C. K. I. Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[15] M. Seeger, C. K. I. Williams, and N. Lawrence. Fast forward selection to speed up sparse Gaussian process regression. In *Proc. Artificial Intelligence and Statistics*, 2003.

[16] E. Snelson and Z. Ghahramani. Sparse Gaussian processes using pseudo-inputs. In *Advances in Neural Information Processing Systems*, Cambridge, MA, 2006. MIT Press.

[17] P. Sollich and C. K. I. Williams. Using the equivalent kernel to understand Gaussian process regression. In *Advances in Neural Information Processing Systems*, Cambridge, MA, 2005. MIT Press.

[18] D. Sun and W.-K. Cham. Postprocessing of low bit-rate block DCT coded images based on a fields of experts prior. *IEEE Trans. Image Processing*, 16(11):2743–2751, 2007.

[19] D. Tschumperlé and R. Deriche. Vector-valued image regularization with PDEs: a common framework for different applications. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 27(4):506–517, 2005.

[20] C. Walder, K. I. Kim, and B. Schölkopf. Sparse multiscale Gaussian process regression. In *Proc. International Conference on Machine Learning*, pages 1112–1119, 2008.

[21] T. Wang and G. Zhai. JPEG2000 image postprocessing with novel trilateral deringing filter. *Optical Engineering*, 47(2):027005–1–027005–6, 2008.

[22] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE Trans. Image Processing*, 19(11):2861–2873, 2010.

[23] Y. Yang, N. P. Galatsanos, and A. K. Katsaggelos. Projection-based spatially adaptive reconstruction of block-transform compressed images. *IEEE Trans. Image Processing*, 4(7):896–908, 1995.

[24] G. Zhai, W. Lin, J. Cai, X. Yang, and W. Zhang. Efficient quadtree based block-shift filtering for deblocking and deringing. *Journal of Visual Communication and Image Representation*, 20(8):595–607, 2009.