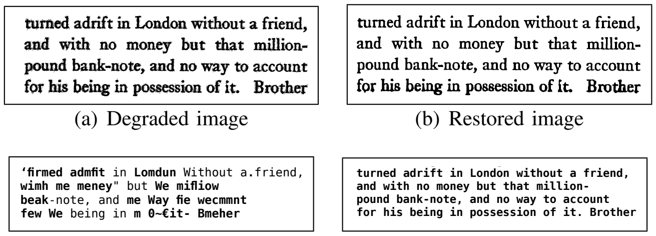
Sparse Document Image Coding for Restoration

Vijay Kumar, Amit Bansal, Goutam Hari Tulsiyan, Anand Mishra, Anoop Namboodiri and C. V. Jawahar

Center for Visual Information Technology, IIIT Hyderabad, India

***Abstract*—Sparse representation based image restoration techniques have shown to be successful in solving various inverse problems such as denoising, inpainting, and super-resolution, etc. on natural images and videos. In this paper, we explore the use of sparse representation based methods specifically to restore the degraded document images. While natural images form a very small subset of all possible images admitting the possibility of sparse representation, document images are significantly more restricted and are expected to be ideally suited for such a representation. However, the binary nature of textual document images makes dictionary learning and coding techniques unsuitable to be applied directly. We leverage the fact that different characters possess similar strokes, curves, and edges, and learn a dictionary that gives sparse decomposition for patches. Experimental results show significant improvement in image quality and OCR performance on documents collected from a variety of sources such as magazines and books. This method is therefore, ideally suited for restoring highly degraded images in repositories such as digital libraries.**

Keywords*—*Document restoration, Sparse representation, Dictionary learning



turned adrift in London without a friend,
and with no money but that million-
pound bank-note, and no way to account
for his being in possession of it. Brother

(a) Degraded image

turned adrift in London without a friend,
and with no money but that million-
pound bank-note, and no way to account
for his being in possession of it. Brother

(b) Restored image

```
'firmed admfit in Lomdun Without a.friend,
wimh me meney" but We mifliow
beak-note, and me Way fie wecmmnt
few We being in m 0~€it- Bmeher
```

(c) OCR output of (a)

```
turned adrift in London without a friend,
and with no money but that million-
pound bank-note, and no way to account
for his being in possession of it. Brother
```

(d) OCR output of (b)

Fig. 1. Part of a scanned page of an old book with severe degradation and the restored image (b). **Bold** words in (c) and (d) indicate differences in Tesseract OCR results. We achieved significant improvement in error rate from 14% to 4.1%.

I. Introduction

Recent years have seen a surge of interest in digitizing the old documents and books to preserve them for posterity and because of their potential applications in information extraction, retrieval etc. Unfortunately many of these old documents and manuscripts are often degraded due to erosion, aging, printing process, ink blot and fading. One such degraded image is shown in Figure 1(a). Apart from cuts and bleeds shown in this example, other types of degradation occur frequently in documents. Restoration may be used as pre-processing step in applications related to recognition and retrieval. Figure 1(c) shows OCR output of Figure 1(a) which is severly affected due to low quality of the document. Clearly, it is necessary to remove these noisy artifacts and restore the degraded document, close to its original form.

Recently, sparse representation has been shown to yield state-of-the-art results in solving inverse problems such as denoising [1][2], inpainting [3] and super-resolution [4], demonstrated on gray and color images, and videos [2]. These works make an assumption that the original clean image of a given degraded image admits a sparse representation with respect to some basis. The sparse codes of the clean image are then recovered from the degraded image. This is due to recent results from compressed sensing [5] that it is possible to efficiently recover a sparse signal from incomplete or noisy measurements provided the basis matrix possess some special properties.

In sparse coding framework, a given signal or image patch is represented as a *sparse* linear combination of an overcomplete basis or dictionary. In this paper, we extend its application for document image restoration that are essentially binary in nature. Our experiments suggest that developing sparse representations for binary images need a slightly different approach than grayscale and color images. We observe that different characters share similar strokes, curves and edges. This allows us to automatically learn a set of features/dictionary that represents them efficiently using the training data. We then seek for high sparsity for degraded images to reconstruct the text regions removing noisy artifacts in documents. Figure 1(b) and (d) shows the result of the degraded image restored by our proposed method and its effect on OCR's performance, respectively. We show an improvement in error rate from 14% to 4.1% in OCR.

Restoration of document images is a well studied topic. There have been many attempts in solving the problem. Gupta *et al.* [6] used a patch based alphabet model to remove blurring artifacts for license plate images using a camera. Lelore *et al.* [7] proposed an approach for the binarization of seriously degraded manuscripts where the MRF model parameters are estimated from the training set. A patch based method is proposed in [8] where each patch is corrected by a weighted average of similar patches, identified using a modified genetic algorithm. Huang *et al.* [9] combined the degradation model and the document model into an MRF framework.

Banerjee *et al.* [10] used an MRF technique that creates an image with smooth regions in both the foreground and the background, while allowing sharp discontinuities across and smoothness along the edges. In their follow-up work [11], they modeled the contextual relationship using an MRF to restore documents with a wide variety of noises. Such methods perform well in restoring many severely degraded documents. However, they have practical limitations from their heavy computational requirements, which increases with larger context.

We briefly look into details of the natural image restoration

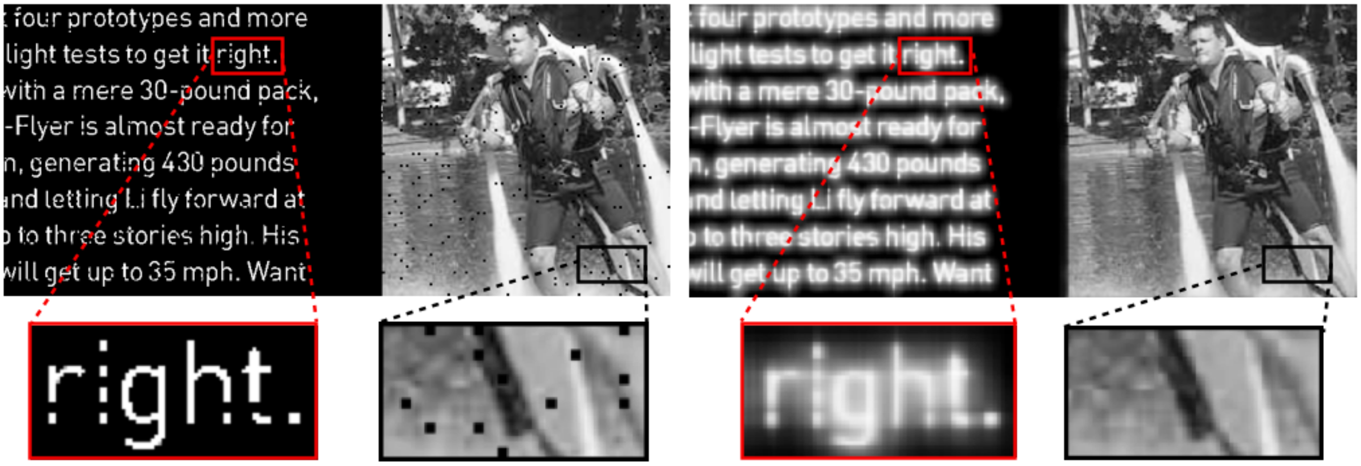

four prototypes and more
light tests to get it right.
with a mere 30-pound pack,
-Flyer is almost ready for
n, generating 430 pounds
and letting Li fly forward at
to three stories high. His
will get up to 35 mph. Want
right.

Fig. 2. Restoration of portion of a magazine (top-left) with text and image for missing pixels and cuts. Region corresponding to natual image is restored well while text region is not. Portions zoomed out with red boxes belong to text and black boxes belong to image. (Best viewed by zooming on a computer)

using sparse representation followed by proposed method for document restoration and experimental results.

II. Sparse coding for Image restoration

In this framework, the task is to recover an image X (clean/high resolution) $\in \mathbb{R}^{M \times N}$ given a degraded (noisy/low resolution/missing values) image Y $\in \mathbb{R}^{M \times N}$. The problem is tackled with the sparsity prior which assumes that natural image patches can be sparsely represented in an appropriately chosen overcomplete basis and their sparse representation can be recovered from the noisy patches. Specifically one assumes that a clean patch $x \in \mathbb{R}^d$ of a clean image X has a sparse representation with respect to an overcomplete basis $D \in \mathbb{R}^{d \times m}$ $(m \ll d)$. i.e,

$$x \approx D\alpha \quad \text{s.t.} \quad ||\alpha||_0 \ll L, \tag{1}$$

where α is the sparse representation of the image patch and $||.||_0$ is l_0 pseudo-norm, which gives a measure of number of non-zero entries in a vector, and the constant L defines the required sparsity level. Finding the sparse solution α is a NP-hard problem. The techniques such as i) greedy methods (matching pursuit [12]) or ii) convex relaxation (l_1-norm) can be used to solve the above problem. Note that, we do not know either the clean image patch x or its representation α. However, we can recover the sparse representation α from incomplete or noisy input image patches y of image Y, with respect to an overcomplete dictionary D due to recent results from [5]. Thus, sparse representation of x is recovered from y as

$$\hat{\alpha} = \min_{\alpha} ||\alpha||_0 \quad \text{s.t} \quad ||y - D\alpha||_2 \leq \epsilon, \tag{2}$$

where ϵ is constant and can be tuned according to the application at hand. For denoising, ϵ could be tuned proportional to noise variance if it is known. As observed in [1][13], learning a dictionary from the images itself instead of a generic basis (DCT or wavelet) could improve the restoration performance.

The above presented sparse coding framework has proved to yield very good results in restoring natural images. However, the application of sparse coding techniques on document restoration is more challenging due to following reasons: (1) Near pixel accurate restorations are important in document images. Errors are immediately visible in binary images as opposed to natural images. (2) Noise in natural images often are uniform and homogenous where the variance is known or estimated, but it is difficult to model the noise in document images. (3) Noise in document images usually contain a mixture of degradations coming from independent processes such as erosion, cuts, bleeds, etc.

We demonstrate the above mentioned challenges with a simple experiment. We consider a portion of the page from a magazine that contains both text and a photograph. We synthetically painted the image with white at randomly selected blocks as shown in Figure 2. Degradation can be treated as missing pixel (inpainting) for photograph and as cuts for text region. We used the sparse coding technique proposed in [3] treating the missing pixels (cuts) as infinite noise and restored the image after learning a dictionary using large number of clean text and natural image patches. It can be seen that the regions corresponding to photograph are restored properly while text regions are not.

III. Restoration of document images

The most critical challenge in restoration of document images using sparse coding can be explained with the help of Figure 3. One of the fundamental assumptions in such a representation is that the elements of the dictionary span the subspace of images of interest and that any linear combination of a sparse subset of dictionary elements is indeed a valid image. This clearly does not hold in the case of document images. Document image patches are binary in nature and so are the dictionary elements (d_i in Figure 3), which is not the case with their linear combination. Ideally, a document image patch (y) that we would like to represent using a dictionary D should be computed as:

$$y = g(D, \alpha), \tag{3}$$

where α is a set of parameters and g is a non-linear function that maps from the binary document dictionary elements to a valid binary document image or patch. Current dictionary learning techniques are not adequate to learn an appropriate dictionary and parameters under such a non-linear mapping.

An alternative is to use a non-linear function (thresholding is a not-so-good example) over a learned linear mapping to a point y' in the subspace.

$$y' = D\alpha, \text{ and } y = f(y') \tag{4}$$

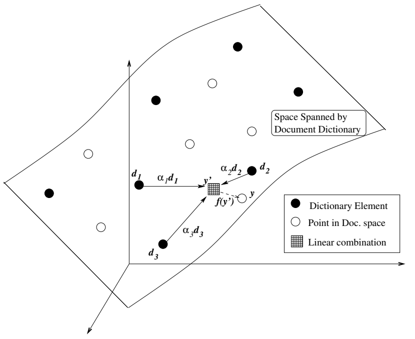


Fig. 3. Space of images and the subspace spanned by the basis vectors in the dictionary (shown in solid black). The document images (white circles) are often outside the subspace spanned by the basis vectors.

Here we approximate the ideal non-linear representation function, g, as $f(y')$, where y' is a linear combination of dictionary elements weighted by α as shown in Figure 3. As seen in Figure 2, the results of such approximations are often very noisy. We get over this problem by approximating a given noisy image using highly sparse representation, where the sparsity is specified to be 1. This ensures that the resulting approximation is close to both binary and a valid document patch.

Our restoration method is follows:

1) Learn a set of representative basis elements that summarizes a given set of clean image patches.
2) Find the sparse representation of each degraded patch over the learned basis and binarize the output.

A. Dictionary Learning

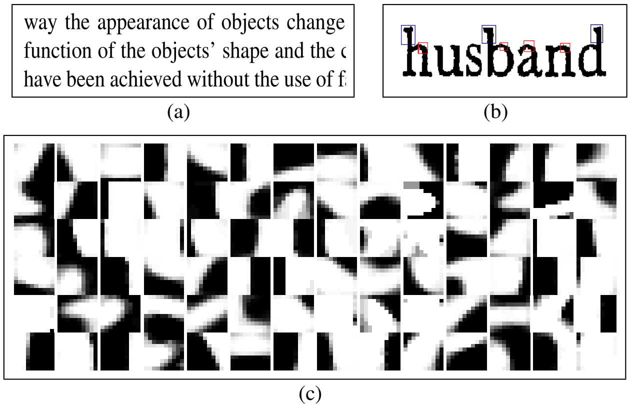
The dictionary learning starts with a set of clean patches extracted from the segmented words. Each word image of size $m \times n$ is split into patches of size $d = p \times q$ resulting in P patches and each patch is represented as a vector $\in \mathbb{R}^d$. For basis learning, we use a method similar to the K-SVD algorithm presented in [1]. We learn the basis $D \in \mathbb{R}^{d \times k}$, such that each patch is represented by a single basis element, as shown in Equation (5). Single non-zero constraint of the coefficients simplifies the K-SVD algorithm to K-means algorithm, however, with the constraint that basis elements are normalized.

$$\{\hat{D}, \hat{\alpha}_i\} = \arg\min_{D,\alpha_i} \sum_{i=1}^{P} ||x_i - D\alpha_i||^2 \tag{5}$$

$$\text{s.t} \quad ||\alpha_i||_0 = 1, \quad \forall i = \{1, \ldots, P\}$$

$$\text{and} \quad ||D_j||_2 = 1, \quad \forall j = \{1, \ldots, k\}$$

The above equation is optimized in an iterative fashion minimizing the objective function over D and α_i, similar to the algorithm presented in [1]. When D is fixed, $\alpha_i \in \mathbb{R}^k$ is given by $\alpha_i^j = D_j^T x_i$ for $j = l$, where $l = \arg\max_l D_l^T x_i$, and $\alpha_i^j = 0$ for $j \neq l$. This would result in the selection of the basis with maximum correlation with the given signal as the representation. Then, each column of D is updated using SVD while fixing other columns, similar to K-SVD algorithm.


way the appearance of objects change
function of the objects' shape and the c
have been achieved without the use of f

(a)

(b)

(c)

Fig. 4. (a) Portion of document page (b) Characters share common strokes, curves, tips, etc. Boxes shown in blue share common tips and boxes shown in red share similar curve (c) Dictionary/Features captures the characteristic information of a text

Figure 4 shows a subset of basis elements learned from a set of patches extracted from word images. Document images are usually binary in nature with 0's and 1's corresponding to text and background regions respectively. Since the interest region is text, we operate on the inverted images to allow the regular conventions of natural image representation. Basis elements learned for document images can be easily interpreted unlike natural images. The fundamental elements that constitute the documents are strokes, curves, glyffs, etc. and our method automatically learns these elements. This can be seen in Figure 4 that dictionary elements correspond to character strokes, thick edges, curves, etc occuring in textual characters (Figure 4 (c)) thereby representing them efficiently. Different English characters possess similar kind of edges, strokes or curves and such patches may share the same dictionary element.

B. Sparse coding

Once the basis is learnt from a set of clean patches, any degraded patch $y_i \in \mathbb{R}^d$ of a noisy image $Y \in \mathbb{R}^{M \times N}$ can be decomposed sparsely over the basis and can be reconstructed as per Equation 5. In order to avoid blocky artifacts in the reconstructed image, we use overlapping patches for restoration and the final reconstructed image is obtained by performing averaging at the overlapped regions.

The reconstructed image might be grayish with little noisy artifacts. Regions corresponding to text will have large pixel values as they are efficiently reconstructed while noisy regions will have small values. We thus use a simple post-processing step that binarizes the gray scale image to remove some of the noisy stray pixels. We found that threshold parameter did not vary too much the quality of the outputs and is fixed to 0.3 in all our experiments.

IV. Experiments and Results

Restoration using the proposed method were carried out on a variety of document images with different levels of degradation. We assume that clean document images with similar font as the one used in the degraded images are available. We also note that our method is robust to slight variation of font between training and testing, as will be demonstrated later. This kind of setting is very much suitable to restore documents from digital libraries, magazines, etc. In such a case, fonts and texts are constant throughout the book and any recent publication of the magazine can be used as high quality training documents. Also, with the advent of internet, one can obtain clean documents with any font easily e.g simple search of 'gothic text' will result in lot of high quality documents which can be used to restore gothic texts. For all the experiments, we segment the image into degraded words and carry out restoration of the individual words.

For learning step, we collected clean documents from a high quality book that has similar font as that of degraded image. Figure 4(a) shows a small region of the clean images collected from a high quality book. The number of sparse coding and basis learning iterations was fixed empirically to 200. In order to maintain overcompleteness and recover sparse representation [5], size of dictionary is usually fixed to four times the size of the patch. It is observed in [1], [13], [3] that very large dictionary leads to overfitting i.e, learnt atoms may correspond to individual patches instead of generalizing for large number of patches and very small dictionary leads to underfitting. Figure 4(b) shows basis elements learned from clean images for a patch size of 15×15.

Figure 5 shows eight different words from the book containing cuts, erosion artifacts, and ink bleed, along with our restoration results. One kind of degradation that we notice is smear and ink blobs, as seen in words *golf*, *fascinating*, *to catch* and *laboratory*. Our algorithm is able to restore these words very well, especially the word *fascinating* which is heavily degraded with characters almost getting connected. Another kind of degradation is fading resulting in near cuts as seen in character *a* in word *sanguinary*, which is restored with high resolution. Our algorithm takes about 12 seconds to restore a document of size 157×663 on a 2GB RAM and Intel(R) Core(TM) $i3-2120$ system with 3.30 GHz processor with un-optimized implementation.

The algorithm however fails to restore the characters *v* and *e* in word *several*. The cut in *v* is very large compared to the size of patch and the horizontal region in *e* has lot of missing pixels and any patch considered in the region is blank and hence algorithm could not estimate the shape in these regions. Similarly, character *y* in *surely* has large amount of bleed which the algorithm failed to restore.

We note that the patch size we considered for restoration has a clear effect on the quality of restoration. If the patch size is too small and comparable with the size of artifacts such as blobs and cuts, the algorithm will restore the noisy region as well. If the patch size is large, the dictionary elements may overfit the training data, resulting in reduced flexibility of degraded images that can be restored. We fixed the size of patches to one-third of character font size.

We evaluate our algorithm both qualitatively and quantitatively on various kinds of synthetically generated degradations such as pixel flipping, blurring, cuts, and texture-blending. An example for each type of degradations and their restored outputs are shown in Figure 7. Flipping is generated using the method proposed in [14] for various PSNR values, by tuning the parameters $\alpha_0, \beta_0, \alpha_1, \beta_1$. Blurring is produced by convolving image with a Gaussian kernel of various sizes. Various levels of cuts are produced by randomly selecting windows in the image and randomly flipping few pixels in the window. Finally, texture-blending simulates effects such as textured paper or stained paper, and was produced by linearly blending the document with a texture image for various degrees of blending. Table 1, shows the input and output PSNR for different kinds of degradations with various levels of noise. We can see a clear improvement in the PSNR values for various degradations.

TABLE I. PSNR (dB) RESULTS OF RESTORATIONS OUTPUTS OF VARIOUS SYNTHETIC DEGRADATIONS.

Flips	Blur	Cuts	Texture blending
6.61 / 6.7	5.1 / 6.9	5.56 / 6.69	4.06 / 6.73
6.75 / 6.82	5.9 / 7.32	6.18 / 8.28	4.47 / 6.77
6.77 / 6.85	6.15 / 7.60	6.96 / 9.01	4.56 / 6.82
6.78 / 6.91	7.05 / 8.40	7.82 / 9.32	4.66 / 6.85

We will now look at the effect of our document restoration on OCR recognition which gives a good measure on the quality of restoration. We used the ABBYY FineReader [15] and Tesseract-2.01 OCR [16] which are the most popular and accurate OCRs available. We ran the OCR on 20 pages of an old English book collected from digital library. Each page of the book contains an average of 300 words and 2200 characters. The error rate measured on degraded documents using ABBYY FineReader was 9% which was already very good. However, after the restoration, it got further reduced to 0.7% which is a significant improvement. Similarly, it got reduced from 14% to 4.1% using Tesseract.

Figure 1 shows the restoration result of a region of degraded page collected from a digital library. Figure 1(c) and (d) show the results of Tesseract OCR output before and after restoration respectively. The recognition error on the degraded page was due to erosion and low printing quality, which might possibly confuse the OCR when the noise fills up the gap between two characters in a word. However, after restoration Figure 1(b), it is recognized with high accuracy.

Figure 6 gives the restored image for the word *played* using popular methods such as median, Gaussian, Non-local means and ours. Clearly, our result is superior in quality compared to these methods. Our method does not make any assumption of script and thus same approach can be applied to restore documents with any script. However, this is beyond the scope of this paper.

V. Conclusion

We present an approach to document restoration, that uses the fact that different characters in a document share similar strokes, curves, edges, etc. We extend the sparse coding based restoration for document images and learned a set of dictionary elements that gives highly sparse decomposition for image patches. We restored severe degradations, including cuts, merges, blobs and erosions in documents, and showed the experimental results on both positive and negative cases. We also demonstrated the improvement in recognition performance of OCR system. Though we demonstrated the application of

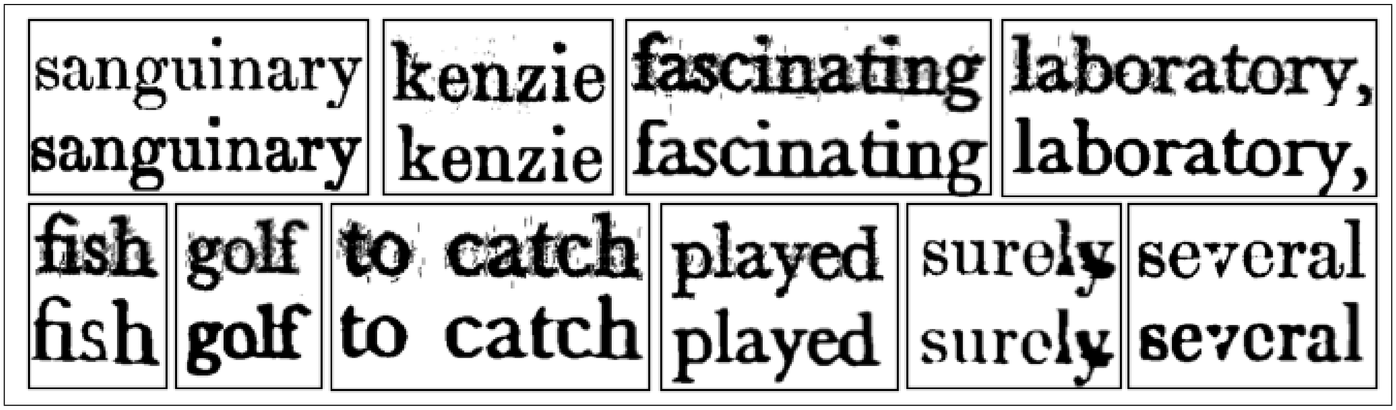
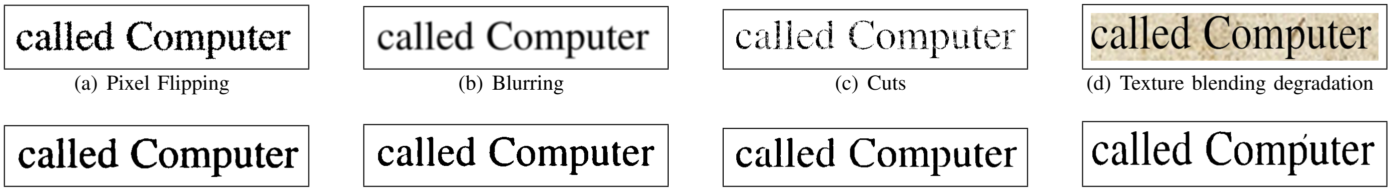
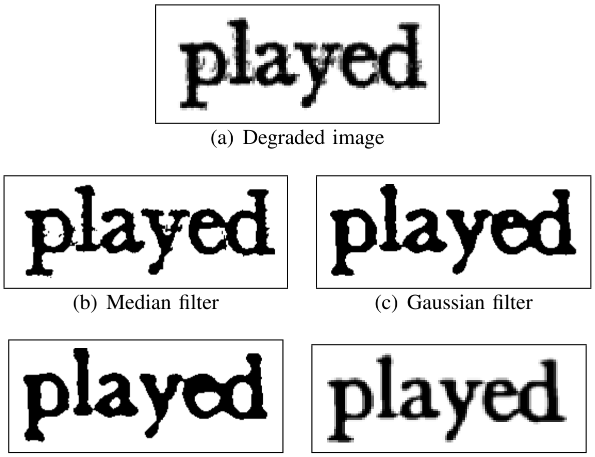


Fig. 5. Restoration of various degraded words. Our algorithm can effectively restore pixel flips, background noise and ink blots (first eight words), while large blobs and cuts that are similar in size to the dictionary patches are not restored (see the last two words).



(a) Pixel Flipping (b) Blurring (c) Cuts (d) Texture blending degradation

Fig. 7. Different kind of synthetic degradations. In each column top image shows degraded image and bottom one shows corresponding restored images. (Best viewed by zooming on computer)



(a) Degraded image

(b) Median filter (c) Gaussian filter

(d) NL-means (e) Ours

Fig. 6. Comaprison with other methods. (a) Cropped word "played" from a degreded document. Output of (b) Median filter (c) Gaussian filter (4) Non-local means (d) Ours. We observe that our restoration technique produces cleaner image as compared to the traditional filtering techniques as well as Non-local means filtering.

sparse coding on challenging document restoration, there is a room for improvement. Unlike natural images, binary images take only few values of intensity and are structured. We would like to work on this aspect along with theoretical guarantees of sparse coding on document images as a part of our future work.

Acknowledgment

This work is partly supported by the MCIT, New Delhi. Vijay Kumar is supported by TCS Reserach PhD fellowship. Anand Mishra is supported by Microsoft Research India PhD fellowship 2012 award.

References

[1] M. Aharon, M. Elad, and A. Bruckstein, "K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation." *IEEE Trans. Singal Process.*, 2006.

[2] J. Mairal, G. Sapiro, and M. Elad, "Learning multiscale sparse representations for image and video restoration," *Multiscale Modeling and Simulation*, 2008.

[3] J. Mairal, M. Elad, and G. Sapiro, "Sparse Representation for Color Image Restoration," *IEEE Trans. Image Process.*, 2008.

[4] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *TIP*, 2010.

[5] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, 2006.

[6] M. D. Gupta, S. Rajaram, N. Petrovic, and T. S. Huang, "Restoration and recognition in a loop," in *CVPR*, 2005.

[7] T. Lelore and F. Bouchara, "Document image binarisation using markov field model," in *ICDAR*, 2009.

[8] R. F. Moghaddam and M. Cheriet, "Beyond pixels and regions: A non-local patch means (nlpm) method for content-level restoration, enhancement, and reconstruction of degraded document images," *PR*, 2011.

[9] Y. Huang, M. S. Brown, and D. Xu, "A framework for reducing ink-bleed in old documents," in *CVPR*, 2008.

[10] J. Banerjee and C. V. Jawahar, "Super-resolution of text images using edge-directed tangent field," in *DAS*, 2008.

[11] J. Banerjee, A. M. Namboodiri, and C. V. Jawahar, "Contextual restoration of severely degraded document images," in *CVPR*, 2009.

[12] S. Mallat and Z. Zhang, "Matching pursuit with time-frequency dictionaries," *IEEE Trans. Singal Process.*, vol. 41, no. 12, 1993.

[13] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *TIP*, 2006.

[14] T. Kanungo, R. M. Haralick, H. S. Baird, W. Stuetzle, and D. Madigan, "A statistical, nonparametric methodology for document degradation model validation," *PAMI*, 2000.

[15] "http://www.abbyy.com/."

[16] "http://code.google.com/p/tesseract-ocr/."