We propose a \(l_0\) sparsity based approach to remove additive white Gaussian noise from a given image. To achieve this goal, we combine the local prior and global prior together to recover the noise-free values of pixels. The local prior depends on the neighborhood relationships of a search window to help maintain edges and smoothness. The global prior is generated from a hierarchical \(l_0\) sparse representation to help eliminate the redundant information and preserve the global consistency. In addition, to make the correlations between pixels more meaningful, we adopt Principle Component Analysis to measure the similarities, which can be both propitious to reduce the computational complexity and improve the accuracies. Experiments on the benchmark image set show that the proposed approach can achieve superior performance to the state-of-the-art approaches both in accuracy and perception in removing the zero-mean additive white Gaussian noise.

1. Introduction

Image denoising, aiming at generating clean images by removing the noises, plays an important role in various tasks of image processing and computer vision, such as feature extraction, object detection, and pattern recognition [1–3]. Since noise-free image can significantly improve the performance of these tasks, image denoising has drawn a growing number of attentions in recent years. Under dissimilar circumstances, images can be corrupted by different types of noises, for example, Gaussian noise, salt-and-pepper noise, and quantization noise, in the process of acquisition and transmission. Among these noises, additive white Gaussian noise, which is usually caused by high temperature or lacking of illumination, is the most conventional and has been widely studied in the past decades. In this paper, we mainly focused on removing this classic type of noise.

Given an image \(V\) corrupted by additive white Gaussian noise \(N\), we can formulate the relationships among \(V\), \(N\), and noise-free image \(U\) as follows:

\[
V = U + N,
\]

where the mean and standard deviation of \(N\) are 0 and \(\delta\), respectively. To recover \(U\) from \(V\) in (1), a number of approaches [4, 5] have been proposed and achieve satisfying results.

Recently, Nonlocal Means (NLM) filtering has been rapidly developed and widely used in image processing tasks. NLM, originally introduced by Buades et al. in [6], tries to utilize all the other pixels in the image to recover the noise-free value of target pixel. NLM has achieved a superior performance [7–9] over previous approaches while preserving the integrity of relevant image information and has been theoretically proven to be effective in dealing with zero-mean Gaussian white noise. In order to globally generate effective correlations among pixels in the target image, the traditional NLM algorithm needs to calculate every pairwise similarity in a large search window. It causes high computing complexity and limits the application of NLM in practice. Therefore, there is an urgent requirement to develop an algorithm to reduce the search range and efficiently construct global correlations among pixels.

Recent studies about the Compressed Sensing (CS) have shown that the information of a given image is always

\[\text{Hindawi Publishing Corporation}
\text{Advances in Multimedia}
\text{Volume 2015, Article ID 386134, 9 pages}
\text{http://dx.doi.org/10.1155/2015/386134} \]
redundant, and a specific pixel can be sparsely represented by other pixels in the image. This inspires us to reduce the search range of NLM by finding the sparse representation of the target pixel. Sparse representation has been used in many computer vision and image processing tasks, such as face recognition, motion segmentation, and image restoration. Specially, \( l_0 \) sparsity based approaches have been well studied, because of its superior performance on various tasks. Xu et al. [10] have utilized \( l_0 \) sparsity to solve the image smoothing and image deblurring problem. Wang et al. [11] adopt \( l_0 \) sparsity in image segmentation. Nie et al. [12] propose a hierarchical \( l_0 \) sparsity based approach to tackle the intrinsic image decomposition problem and achieve superior performance to the previous approaches. These successful applications of \( l_0 \) sparsity inspire us to explore the potentiality of it to resolve image denoising problem.

In this paper, we propose a \( l_0 \) sparsity based approach to remove the additive white Gaussian noise from a given image. To recover the noise-free value of a pixel, we combine the local and global information, together, to achieve more accurate result. The local correlation is adopted by exploring the relationships of pixels in a small search window (3 \times 3 pixels in our experiments) to preserve edges and ensure local smoothness. The global correlation is generated by finding the sparse representation based on \( l_0 \) sparsity to reduce the search range. Moreover, we also adopt Principal Component Analysis (PCA) to measure the similarities between features of pixels, which can both reduce the amount of calculation and improve the accuracies. Experiments on benchmark images corrupted by different level of zero-mean Gaussian white noise show that the proposed approach gains superior performance to the state-of-the-art image denoising approaches.

The rest of this paper is organised as follows: Section 2 will briefly review the NLM algorithm and \( l_0 \) sparsity. Section 3 will specify the proposed \( l_0 \) based nonlocal denoising algorithm in detail. In Section 4, we show the experimental results that demonstrate the effectiveness of our algorithm. Section 5 will summarise and conclude this paper.

2. Related Work

In this section, we will briefly review the previous studies about NLM and \( l_0 \) sparsity, the two most correlated algorithms to the proposed image denoising approach.

Nonlocal Means. NLM is a popular technique developed by Buades et al. [13], which achieves unprecedented performance in image denoising task. However, the superior performance is achieved at the cost of high computational complexity. For a target pixel \( p \), the standard NLM computes the relevances based on image patches between \( p \) and all the other pixels in the image. Then, the linear combination result of similar and homologous pixel values is assigned as the noise-free value of \( p \). The high computational complexity makes standard NLM less practical in image denoising task. Salmon [14] conducted experiments to study two factors in NLM, search window size and weight of the central patch, and concluded that the performance increased when the window is larger and the way of the central weight was a crucial issue. Some approaches have been proposed to accelerate the NLM or find a more appropriate definition of weight. References [15, 16] terminated the distortion computation between patches based on probabilistic early termination. But the reduction in the amount of similarity measurement resulted in the loss of image information. Darbon et al. [17] separated the pixels in neighbourhood to offer a parallel implementation in contemporary memory sharing of computer architectures, which had special requirement for hardware. In terms of weight, Zhong et al. [18] estimated the similarities between noise-free patches instead of noisy observations. The denoising scheme did secondary filtering based on the original NLM, which resulted in more calculation amount. References [19, 20] built a dictionary to project the image content to the subspace and then defined a metric between a pixel and the neighbouring ones. According to region characteristics, region-based NLM [21, 22] adaptively changed the similarity patch size to preserve edges. These methods did provide a more accurate metric of distance and gained a better performance than the standard NLM. However, because the search area is a region centered on the target pixel, the previous NLM based approaches are seminonlocal in essence.

\( l_0 \) Sparsity. \( l_0 \) sparsity based approaches have been widely used in many vision and graphic tasks and acquired superior performance. Hyder and Mahata [23] proposed an iteratively approximate \( l_0 \)-norm based on fixed point to reconstruct sparse signal and achieved an obvious improvement in noisy environment. In [10, 24], Xu et al. developed \( l_0 \) gradient minimization to control the nonzero gradients among neighborhoods, which indicated that the prominent image structures needed to be preserved in image smoothing. Later, they extended \( l_0 \) sparsity to image deblurring and achieved notable improvement both in convergence speed and result quality. Mancera and Portilla [25] selected coefficients by minimizing the nonzeros with \( l_0 \)-norm in inpainting problem and gained a remarkable performance compared to practical \( l_1 \)-norm based methods. A Gaussian function was provided by Mohimani et al. [26] to obtain a continuous smooth estimation of \( l_0 \)-norm and found the sparsest solution for linear equations of an underdetermined system. The process speed was improved by two orders of magnitude faster than \( l_1 \) while retaining the same accuracy. Lopez et al. [27] adapted \( l_0 \)-norm to make Least-Squares Support Vector Machines (LS-SVM) sparsity in classification and regression problems. As a result, the amount of support vectors was reduced significantly than standard LS-SVM while keeping a comparable accuracy. However, this procedure was computationally expensive. Wang et al. [11, 28] built a constructed graph based on \( l_0 \) representation of features, which gave a better description of superpixels, to perform image segmentation. Experimental results showed that their approach achieves competitive results compared with state-of-the-art methods. Nie et al. [12] constructed a sparse representation for pixels by solving the \( l_0 \) minimization problem, then formulated a sparse prior to preserve the global consistency during the decomposition of intrinsic image, and
achieved superior performance. The success of $l_0$ sparsity in various tasks promotes us to utilize it to deal with image denoising problem.

3. Our Method

3.1. The Proposed Formulation. Let $U$, $N$, and $V$ represent noise-free image, additive Gaussian noise, and observed image, respectively. According to (1), the value of pixel $i$ in $V$ can be defined as the addition of $i$th pixel values in $N$ and $U$; that is, $V(i) = U(i) + N(i)$. We formulate the proposed denoising model as follows:

$$
\hat{U}(i) = \frac{1}{Z(i)} \left( P_l(i) + P_g(i) \right),
$$

where $\hat{U}(i)$ is the restored value of pixel $i$, $P_l(i)$ and $P_g(i)$ represent the local prior and global prior at pixel $i$, and $Z(i)$ is a normalization factor.

Local prior $P_l$ explores the correlations among pixels in a local window. It can preserve edges and encourage smoothness. We define the local prior at pixel $i$ as

$$
P_l(i) = \sum_{m \in N(i)} w_l(i, m) V(i),
$$

where $N(i)$ represents the pixel set in a local search window centred at $i$ and $w_l(i, m)$ represents the local similarity between pixel $i$ and pixel $m$. In our experiments, the size of local search window is set as $3 \times 3$. We will illustrate the detailed definition of $w_l$ in the next section.

Global prior $P_g$ builds the relationships between pixels in the whole image. It can promote the global consistency and improve the denoising results. The definition of $P_g$ is given as follows:

$$
P_g(i) = \sum_{n \in N_g(i)} w_g(i, n) V(i),
$$

where $N_g(i)$ represents the set of pixels globally correlated with $i$ and $w_g(i, n)$ represents the nonlocal similarity. The generation of $N_g(i)$ and calculation of $w_g(i, n)$ will be specified in Section 3.3.

The normalization factor $Z(i)$ is defined as

$$
Z(i) = \sum_{m \in N(i)} w_l(i, m) + \sum_{n \in N_g(i)} w_g(i, n).
$$

3.2. Local Prior Acquisition. According to (3), the local prior is defined as the summation of weighted pixel values in a local search window. In this section, we will specify the definition of similarity $w_l(i, m)$.

For a given pixel $i$, we represent its feature by concatenating the gray intensity of nearby pixels within a local square patch centred at pixel $i$, and the size $S$ of the patch is set as 5 in our experiments. We use $\mathcal{F} = \{f_i\}$ to denote the feature set of all pixels in image $V$, where $f_i$ is the feature vector of pixel $i$ and its dimension is $S^2$. For pixel $m \in N_g(i)$, the similarity between pixels $i$ and $m$ is defined as follows:

$$
w_l(i, m) = \exp \left( -\frac{\|f_i - f_m\|^2}{\sigma^2} \right),
$$

where $\| \cdot \|_2$ represents the Euclidean distance and $h$ is the parameter which controls the decay of weights related to the level of noise. For a specific image, $h$ is a constant. The similarity definition given in (6) can help maintain the simplicity of algorithm; however, the computation process is time-consuming. In addition, the original feature vector always includes redundant and noisy information. To reduce the feature dimension and refine the feature representation, we project the original feature vector onto a subspace generated by Principle Component Analysis (PCA).

To achieve this goal, we first randomly select $M$ pixels in image $V$. We use $\mathcal{F}_{\text{sub}} = \{f_i\}^M$ to denote the set of feature vectors of selected pixels. Let $K$ be the number of principle components and let $\{\lambda_j\}_{j=1}^K$ be the sorted eigenvectors in descending order according to their respective eigenvalues, which are generated from the set $\mathcal{F}_{\text{sub}}$. The number $K$ is determined by

$$
\frac{\sum_{j=1}^K \lambda_j}{\sum_{j=1}^M \lambda_j} \geq \theta,
$$

where $\theta$ is a threshold which means the utilization rate of information, usually set as 0.95 in our experiments. Then, we project the original feature vector $f_i$ on the $K$-dimensional PCA subspace as

$$
\tilde{f}_i = \sum_{j=1}^K \langle f_i, \lambda_j \rangle \cdot \lambda_j,
$$

where $\langle \cdot \rangle$ represents the distance of $i$th original feature vector's projection on the $j$th basis vector. Since the basis vectors $\{\lambda_j\}_{j=1}^K$ are orthonormal, we can calculate the distance between pixel $i$ and pixel $m$ on the PCA subspace as

$$
\text{Dist}(i, m) = \sum_{j=1}^K \left( \langle f_i, \lambda_j \rangle - \langle f_m, \lambda_j \rangle \right)^2.
$$

We use Dist$(i, m)$ to replace the traditional Euclidean distance. Finally, the local similarity between pixels $i$ and $m$ is defined as

$$
w_g(i, m) = \exp \left( -\frac{\text{Dist}(i, m)}{\sigma^2} \right).
$$

3.3. Global Prior Acquisition. Based on the mechanism of sparse coding representation, a specific signal $s$ can be sparsely represented by the dictionary $D$ with the coefficient vector $\alpha$, that is, $s = D\alpha$, subject to $|\alpha| \leq \tau$, and $\tau$ is a positive integer. Accordingly, for a given pixel $i$, we still use $f_i$ to represent its feature vector. To generate the global correlations of pixel $i$, we solve the following minimization problem based on $l_0$ sparsity:

$$
\min_{\alpha_i} \|f_i - D\alpha_i\|_2^2
$$

s.t. $\|\alpha_i\|_0 \leq \tau,
$$

where $\alpha_i$ represents the feature coefficients.
where $D_i$ represents the sparse dictionary of pixel $i$ and $\alpha_i$ is the corresponding coefficient vector. The construction of dictionary $D_i$ for each pixel can significantly affect the accuracy and speed of the solving process. Inspired by the hierarchical $l_0$ sparsity proposed in [12], which has shown superior performance both in effectiveness and efficiency, we also adopt a hierarchical framework in this paper.

For image $V$, given the downsampling ratio $r$ and layer number $T$, we first build an image pyramid $\mathcal{P}_V = \{V^t\}_{t=1}^T$, $V^t$ represents the image in $t$th layer, and $V^{T'}$ corresponds to the original image $V$. For the top layer $V^1$, we build the sparse dictionary $D^1_i$ for pixel $i$ by features of all the other pixels in $V^1$, that is, $D^1_i = [f^1_i, f^1_{i-1}, f^1_{i+1}, \ldots, f^1_L]$, where $f^1_i$ represents the feature of $i$th pixel in $V^1$ and $L^1$ is the number of pixels in $V^1$. By solving the coefficient vector $\alpha^1_i$, we can find pixels which are globally correlated with pixel $i$ through the nonzero coefficients in $\alpha^1_i$. For layer $V^t (1 < t \leq T)$, we build sparse dictionary $D^t_i$ for $i$th pixel by the derivative relationships from $V^{t-1}$. Specifically, we first find pixel $i$’s corresponding pixel $i'$ in $V^{t-1}$ by downsampling. By $\alpha^t_{i'}$, we can find the pixel set $\mathcal{X}^t_{i'}$ correlated with $i'$. For each pixel $j \in \mathcal{X}^t_{i'}$, we can get the corresponding image patch $Y^t_j$ in $V^t$ by upsampling. Then, we use feature vectors of pixels in $\bigcup_{j \in \mathcal{X}^t_{i'}} Y^t_j$ to construct the sparse dictionary $D^t_i$ for $i$th pixel in layer $t$, where $\bigcup$ represents the union of multiple sets.

Finally, we can find the set $\mathcal{N}_g(i)$ of pixels which are globally correlated to pixel $i$ in $V$ by nonzero coefficients in $\alpha^T_i$. For pixel $n \in \mathcal{N}_g(i)$, the global similarity between pixels $i$ and $n$ is defined by the Normalized-Residual as

$$w_g(i, n) = \frac{\|f_i - D_i\alpha_i^n\|_2^2}{\|f_i\|_2^2}, \quad (12)$$

where $\alpha_i^n(z) = 0$, if $z = n$, and $\alpha_i^n(z) = \alpha_i(z)$, otherwise.
4. Experimental Result

We test our approach on twelve benchmark images: Lena, Boat, Barbara, Man, Cameraman, House, Peppers, Couple, Hill, Fingerprint, Mandrill, and Monarch. Images are corrupted by three levels of additive zero-mean white Gaussian noise, with standard variance $\sigma = 20, 30, 40$, respectively. Parameters to achieve local prior are set as illustration in Section 3.2. For parameters in global prior acquisition, we set the sparsity $r = 5$, the downsampling ratio $r = 2$, and the layer number $T = 4$. The parameter settings are generated by fine-tuning on the training set, including Montage, Airplane, Bridge, and Truck, to achieve the highest average accuracy. We quantitatively evaluate the denoising quality with two conventional measurements: Peak Signal to Noise Ratio (PSNR) and Structural Similarity Index Measure.
Table 1: PSNR/SSIM of denoising images using different models with 3 noise levels.

<table>
<thead>
<tr>
<th></th>
<th>NLM</th>
<th>SIM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SIM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SIM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SSIM</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lena</td>
<td>31.53</td>
<td>0.89</td>
<td>31.35</td>
<td>0.91</td>
<td>31.52</td>
<td>0.90</td>
<td>31.90</td>
<td>0.92</td>
<td>32.04</td>
<td>0.91</td>
<td>32.08</td>
<td>0.92</td>
<td>32.08</td>
<td>0.92</td>
<td>32.08</td>
<td>0.92</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Boat</td>
<td>29.30</td>
<td>0.85</td>
<td>29.67</td>
<td>0.90</td>
<td>29.45</td>
<td>0.76</td>
<td>29.60</td>
<td>0.88</td>
<td>29.92</td>
<td>0.88</td>
<td>29.95</td>
<td>0.90</td>
<td>29.95</td>
<td>0.90</td>
<td>29.95</td>
<td>0.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Barbara</td>
<td>30.09</td>
<td>0.92</td>
<td>28.77</td>
<td>0.91</td>
<td>28.59</td>
<td>0.82</td>
<td>30.37</td>
<td>0.93</td>
<td>29.76</td>
<td>0.93</td>
<td>30.16</td>
<td>0.93</td>
<td>30.16</td>
<td>0.93</td>
<td>30.16</td>
<td>0.93</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Man</td>
<td>28.48</td>
<td>0.87</td>
<td>29.53</td>
<td>0.89</td>
<td>29.21</td>
<td>0.86</td>
<td>29.57</td>
<td>0.87</td>
<td>29.26</td>
<td>0.89</td>
<td>29.64</td>
<td>0.90</td>
<td>29.64</td>
<td>0.90</td>
<td>29.64</td>
<td>0.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cameraman</td>
<td>29.00</td>
<td>0.77</td>
<td>29.60</td>
<td>0.85</td>
<td>29.42</td>
<td>0.72</td>
<td>29.69</td>
<td>0.84</td>
<td>30.19</td>
<td>0.82</td>
<td>30.28</td>
<td>0.82</td>
<td>30.28</td>
<td>0.82</td>
<td>30.28</td>
<td>0.82</td>
<td></td>
<td></td>
</tr>
<tr>
<td>House</td>
<td>32.32</td>
<td>0.73</td>
<td>32.20</td>
<td>0.84</td>
<td>31.55</td>
<td>0.81</td>
<td>32.48</td>
<td>0.85</td>
<td>32.69</td>
<td>0.79</td>
<td>32.88</td>
<td>0.86</td>
<td>32.88</td>
<td>0.86</td>
<td>32.88</td>
<td>0.86</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Peppers</td>
<td>29.69</td>
<td>0.80</td>
<td>30.16</td>
<td>0.85</td>
<td>29.70</td>
<td>0.77</td>
<td>30.77</td>
<td>0.87</td>
<td>30.78</td>
<td>0.84</td>
<td>30.87</td>
<td>0.87</td>
<td>30.87</td>
<td>0.87</td>
<td>30.87</td>
<td>0.87</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Couple</td>
<td>29.14</td>
<td>0.90</td>
<td>29.29</td>
<td>0.90</td>
<td>29.46</td>
<td>0.79</td>
<td>29.31</td>
<td>0.88</td>
<td>—</td>
<td>—</td>
<td>29.63</td>
<td>0.91</td>
<td>29.63</td>
<td>0.91</td>
<td>29.63</td>
<td>0.91</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Hill</td>
<td>29.50</td>
<td>0.88</td>
<td>29.66</td>
<td>0.88</td>
<td>29.68</td>
<td>0.83</td>
<td>29.40</td>
<td>0.85</td>
<td>—</td>
<td>—</td>
<td>29.78</td>
<td>0.88</td>
<td>29.78</td>
<td>0.88</td>
<td>29.78</td>
<td>0.88</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fingerprint</td>
<td>26.81</td>
<td>0.97</td>
<td>26.86</td>
<td>0.95</td>
<td>27.30</td>
<td>0.83</td>
<td>27.43</td>
<td>0.95</td>
<td>—</td>
<td>—</td>
<td>27.56</td>
<td>0.95</td>
<td>27.56</td>
<td>0.95</td>
<td>27.56</td>
<td>0.95</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mandrill</td>
<td>25.63</td>
<td>0.89</td>
<td>25.62</td>
<td>0.87</td>
<td>26.07</td>
<td>0.77</td>
<td>25.94</td>
<td>0.86</td>
<td>—</td>
<td>—</td>
<td>26.17</td>
<td>0.89</td>
<td>26.17</td>
<td>0.89</td>
<td>26.17</td>
<td>0.89</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Monarch</td>
<td>29.20</td>
<td>0.86</td>
<td>29.30</td>
<td>0.89</td>
<td>28.63</td>
<td>0.83</td>
<td>29.58</td>
<td>0.90</td>
<td>—</td>
<td>—</td>
<td>29.70</td>
<td>0.90</td>
<td>29.70</td>
<td>0.90</td>
<td>29.70</td>
<td>0.90</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Average</td>
<td>29.22</td>
<td>0.86</td>
<td>29.34</td>
<td>0.88</td>
<td>29.22</td>
<td>0.81</td>
<td>29.67</td>
<td>0.88</td>
<td>—</td>
<td>—</td>
<td>29.89</td>
<td>0.89</td>
<td>29.89</td>
<td>0.89</td>
<td>29.89</td>
<td>0.89</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(SSI M). PSNR represents noise removal effect, defined as follows:

\[
\text{PSNR}(\hat{U}, U) = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right)
\]

(13)

with

\[
\text{MSE}(\hat{U}, U) = \frac{1}{\Omega} \sum_{i=1}^{\Omega} (\hat{U}(i) - U(i))^2,
\]

(14)

where \( \hat{U} \) and \( U \) represent the denoising image and noise-free image, respectively, and \( \Omega \) is the number of pixels in image \( U \).
Figure 3: Detailed comparison of the proposed approach with other approaches on image Lena with $\sigma = 30$: (a) is the noise-free image; (b) is the corrupted image; (c)–(f) are the denoising results by NLM, NL-TV, NLMPCA, and NLMSAP; (g) is the denoising result by the proposed $l_0$-NLM.
Figure 4: Detailed comparison of the proposed approach with other approaches on image Man with \( \sigma = 30 \): (a) is the noise-free image; (b) is the corrupted image; (c)–(f) are the denoising results by NLM, NL-TV, NLMPCA, and NLMSAP; (g) is the denoising result by the proposed \( l_0 \)-NLM.

superior performance of the proposed approach to state-of-the-art approaches in different noise levels.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

The authors would like to extend their sincere gratitude to Professor Dang Jianwu, for his supports and efforts on this work. The research is supported in part by the National Natural Science Foundation (Surface Project no. 61175016 and Surface Project no. 61304250).

References


Advances in Multimedia


Submit your manuscripts at
http://www.hindawi.com