Motivation and Goals

- Poisson noise appears in many imaging applications such as fluorescence microscopy, computed tomography (CT), night vision, etc.
- We aim at recovering such noisy images.
- We propose an algorithm that relies on the Poisson noise statistics and sparsity-based image modeling, leading to state-of-the-art results.

The Poisson Denoising Problem

- \( \mathbf{x}_0 \) and \( y \) - clean and noisy images
- The goal is to recover \( \mathbf{x}_0 \) from \( y \)
- Each element \( y[i] \) in \( y \) is (an integer) i.i.d. Poisson distributed with mean and variance \( \lambda \):
  \[
P(y[i] = m | \mathbf{x}_0[i] = \lambda) = \frac{\lambda^m e^{-\lambda}}{m!}, \quad \lambda > 0
  \]
- Large \( \mathbf{x}_0[i] \) --- large \( y[i] \), small \( \mathbf{x}_0[i] \) --- small \( y[i] \)
- The noise power is measured by the peak value: maximal value in \( \mathbf{x}_0 \)
- In scenarios of low noise/large peak, the Poisson noise looks like an additive i.i.d. Gaussian noise:

\[
x_0 \quad \text{peak} = 100
\]

- The Anscombe transform can be used to convert the Poisson noise into a Gaussian one, and then standard denoising methods can be applied
- In this work we deal with the case of high noise/small peak value (peak<4)

- The Anscombe transform is no more effective and the Poisson noise has to be treated directly

Sparsity Based Reconstruction

- By maximizing the log-likelihood of the Poisson distribution we get the following minimization problem
  \[
  \min \mathbf{x}^T \mathbf{x} - y^T \log(x)
  \]
- A regular sparsity prior \( \mathbf{x} = \mathbf{D}_a, \|\mathbf{a}\|_0 < k \) leads to a non-negative optimization
  \[
  \min \mathbf{x}^T \mathbf{D}_a - y^T \log(D_a) \text{ s.t. } \|\mathbf{a}\|_0 < k, \mathbf{D}_a > 0
  \]
- Instead, similar to [2], we use \( \mathbf{x} = \exp(\mathbf{D}_a), \|\mathbf{a}\|_0 < k \) that leads to
  \[
  \min \mathbf{x}^T \exp(\mathbf{D}_a) - y^T \mathbf{D}_a \text{ s.t. } \|\mathbf{a}\|_0 < k
  \]
- Note that [2] uses a GMM, which is different from our scheme
- \( \mathbf{D} \) is a “given” dictionary and \( |\cdot| \) counts the non-zero elements
- This problem is NP-hard and thus we seek an approximation

Poisson Greedy Algorithm for Sparse Coding

- Input: Group of noisy patches \( \{y_1, \ldots, y_l\} \) such that their original clean versions have the same support in their representation \( \exp(\mathbf{D}_a), 1 \leq i \leq l \) with \( k \) non-zeros
- We find these support by a greedy procedure:
  - Initialization: the support \( T = \{\}, t = 0 \)
  - For \( t = 1 : 1 : k \):
    - Find new support element and representations:
      \[
      \left[ \tilde{\alpha}_1, \ldots, \tilde{\alpha}_l, j^* \right] = \arg \min_{\tilde{\alpha}_1, \ldots, \tilde{\alpha}_l, j} \sum_{i \in T} \exp(D_{t\leftrightarrow i} \tilde{\alpha}_i) - y_i^T D_{t\leftrightarrow i} \tilde{\alpha}_i,
      \]
  - Add new support to \( T \) andToUpdate \( \tilde{\alpha}_i \)

Our Sparse Poisson Denoising Algorithm

1. Divide the image into set of overlapping patches
2. Gaussian filtering – used only for the task of clustering the patches
3. Cluster (using the Gaussian filtered image) the noisy patches into large number of small groups
4. Apply the Poisson greedy algorithm for each group assuming that patches have the same non-zero locations (support) in their representations
5. Form the final image from the reconstructed patches by averaging

A global dictionary \( \mathbf{D} \) is used for all groups

Improvements

- Dictionary learning stage – after we have representations for all the patches we do a Newton step for updating \( \mathbf{D} \)
- Error based stopping criterion: add elements to the support till the error between the reconstructed patches and the patches of the reconstructed image stop decreasing

Recovery Performance

<table>
<thead>
<tr>
<th>Image</th>
<th>Peak(%)</th>
<th>Peak(%)</th>
<th>Peak(%)</th>
<th>Peak(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>BM</td>
<td>10.68</td>
<td>13.38</td>
<td>15.66</td>
<td>19.01</td>
</tr>
<tr>
<td>3</td>
<td>6.98</td>
<td>9.97</td>
<td>11.97</td>
<td>14.97</td>
</tr>
<tr>
<td>1</td>
<td>5.86</td>
<td>8.86</td>
<td>10.86</td>
<td>12.86</td>
</tr>
<tr>
<td>2</td>
<td>6.98</td>
<td>9.97</td>
<td>11.97</td>
<td>14.97</td>
</tr>
<tr>
<td>2(Anscombe)</td>
<td>5.86</td>
<td>8.86</td>
<td>10.86</td>
<td>12.86</td>
</tr>
</tbody>
</table>

References