Motivation

- Acquisition of Image is usually noisy
- Application of denoising algorithms prior to image processing algorithms
- Each denoising algorithm has pros and cons
  - Median Filter
  - Forward/Backward Recursive Algorithm
  - Discrete Universal DEnoising (DUDE)
Algorithm 1: Median Filter

- Non-linear, sliding window technique
- Edge effects
- Example: \([12 \ 55 \ 23 \ 1 \ 7]\) with window size 3
Algorithm 2: Hidden Markov Models

- A Markov Chain:
  \[ P(X, Z \mid Y) = P(X \mid Y)P(Z \mid Y) \]

- Hidden Markov Process:
Algorithm 2: Hidden Markov Models (Continued)

- Forward/Backward recursive algorithm
  - Derivation based on properties of probability densities and markovity
  - Goal: Determine $P(x_t \mid y_{1:t-1})$ for $t = 1, \ldots, n$
1) **Initialization:**
   i. Some initial distribution $P(x_1 \mid y_{1\rightarrow 0}) \doteq P(x_1)$
      $\rightarrow$ Setup to propagate the recursion.

   ii. Markov kernel: $P(x_t \mid x_{t-1})$

   iii. Corruption channel: $P(y_t \mid x_t)$

   iv. Noisy observations: $Y_1, Y_2, Y_3, \ldots$
Algorithm 2: Hidden Markov Models – Forward Recursion

2) Recursive step
For \( t = 1 \) to \( n \) step 1

\[
P(x_t \mid y_{1\rightarrow t}) = \frac{P(x_t \mid y_{1\rightarrow t-1})P(y_t \mid x_t)}{\sum_{\tilde{x}_t} P(x_t \mid y_{1\rightarrow t-1})P(y_t \mid \tilde{x}_t)}
\]

(Measurement update)

\[
P(x_{t+1} \mid y_{1\rightarrow t}) = \sum_{x_t} P(x_t \mid y_{1\rightarrow t})P(x_{t+1} \mid x_t, y_{1\rightarrow t})
\]

(Time update)

End For-loop
Algorithm 2: Hidden Markov Models – Backward Recursion

Goal: Determine $P(x_t \mid y_{1\rightarrow n})$ for $t = 1, \ldots, n$

1 - Initialization:
   i. From the Forward Recursive Algorithm we have
      $P(x_t \mid y_{1\rightarrow t})$
      for $t = 1, \ldots, n$
   
   ii. Markov kernel:
      $P(x_t \mid x_{t-1})$
Algorithm 2: Hidden Markov Models – Backward Recursion

2 - Recursive step
For $t = n-1$ to $1$ step $-1$

$$P(x_t \mid y_{1\rightarrow n}) = \sum_{x_{t+1}} \frac{P(x_t \mid y_{1\rightarrow t})P(x_{t+1} \mid x_t)}{\sum_{x_t} P(x_t \mid y_{1\rightarrow t})P(x_{t+1} \mid x_t)} P(x_{t+1} \mid y_{1\rightarrow n})$$

End For-loop
Algorithm 3: Discrete Universal DEnoising Algorithm (DUDE)

- No assumptions about underlying probability distribution
- Channel statistics known
- A loss function is specified
  - Hamming Loss:

\[ 1 \text{ if } x \neq \hat{x}, \ 0 \text{ if } x = \hat{x} \]
Algorithm 3: Discrete Universal DEnoising Algorithm (DUDE)

Step 1: First pass through the data set – To compute count vectors

\[ m(a^n, b^k, c^k)[\beta] = \left| \{i : k + 1 \leq i \leq n - k, a_{i-k\rightarrow i+k} = b^k \beta c^k \} \right| \]
Algorithm 3: Discrete Universal DEnoising Algorithm (DUDE)

Step 2: Second Pass through the data set – Denoising step

We correct each pixel according to this rule:

$$\arg \min_{\hat{x} \in A} m^T (y^n, b^k, c^k) \Pi^T [\Pi \Pi^T]^{-1} [\lambda_{\hat{x}} \odot \pi_{z_i}]$$

where

$$\Pi = [\pi_1 | ... | \pi_M]$$

$$\Lambda = [\lambda_1 | ... | \lambda_M]$$

$\odot$ represents component-wise multiplication.
Proof-of-Concept

- Application on Bi-level Images
Corruption Mechanism

\[
\Pi = \begin{bmatrix}
1-p & p \\
p & 1-p
\end{bmatrix}
\]
Simplification of DUDE

If $x = 1$

If
\[
\frac{m(y^k, b^k, c^k)[x]}{m(y^n, b^k, c^k)[x] + m(y^n, b^k, c^k)[x]} > 2p(1 - p)
\]
\[\hat{x} = 1\]

Else
\[\hat{x} = 0\]

End If

Else

If
\[
\frac{m(y^k, b^k, c^k)[\bar{x}]}{m(y^n, b^k, c^k)[x] + m(y^n, b^k, c^k)[x]} > 2p(1 - p)
\]
\[\hat{x} = 0\]

Else
\[\hat{x} = 1\]

End if

End if
Results

- Original Image:
Results

- Noisy Image ($p=0.25$):
Results

- Median Filter (Window length = 9):

  \[ \text{post\_error\_rate} = 0.0986 \]
Results

- Forward/Backward Recursive Algorithm
- 3 iterations

post_error_rate = 0.0589
Results

- DUDE (Window length = 9)

post_error_rate = 0.0995
Results

• Performance Analysis

Error Rates for Different Algorithms
- Forward/Backward 3 iterations
- Median Filter window size 9
- DUDE window size 9
Conclusion

- Median filter decent – easy to implement
- Forward/Backward best – but huge storage
- DUDE is pretty amazing given assumptions
  - Has rooms for expansion
    - Eg. Continuous-tone images
Thank You!

- Questions???