Statistical Image Segmentation Expectation Maximization/Maximization of the Posterior Marginals EMMPM

Khalid Tahboub
Purdue University

April, 8th, 2011

PURDUE
ENGINEERING

Presentation outline:

Basic EMMPM model
Incorporating edge information
Incorporating curvature
Markov Chain Markov Random Field (MCMRF) Model

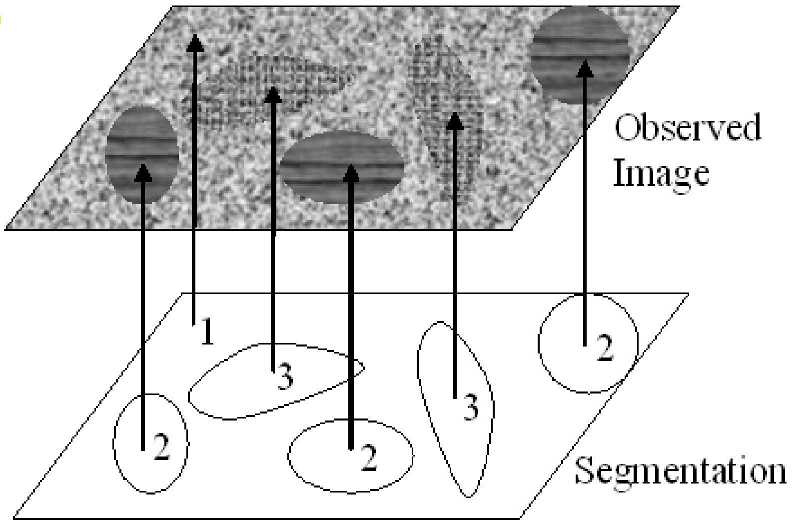
PURDUE ENGINEERING

Basic EMMPM

- Based on the Markov Random Field (MRF) model.
- Very effective for imposing smoothness constraints on the segmentation.
- Pixels that are spatially close are likely to belong to the same class.
- Applied with a relatively simple probability model.
- Ising model for the class labels combined with the assumption that each class produces identical and independent Gaussian samples with mean and variance particular to that class.

General Idea

- Estimate segmentation from observed image
- Use statistical model for observed image
- Use statistical model for true segmentation

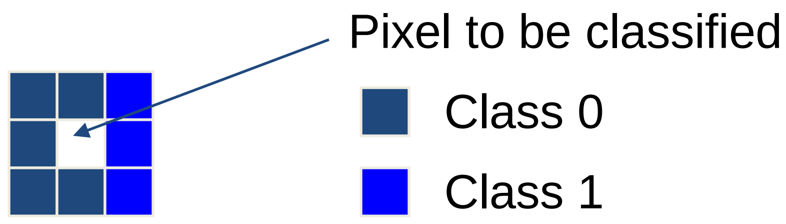


Pr(segm. x|obs. image y) ~ Pr(y|x)·Pr(x)

EM/MPM method finds segmentation that mimimizes expected number of misclassified pixels while estimating parameters of observed image model automatically

Markov Random Field

- Add term to model correlations between neighboring pixels
  - Markov random field model assigns cost at each pixel for being classified differently from neighbors



$$P(\text{Class } 0) \sim \exp\left(-\beta \cdot (\#\text{ of neighbors different from Class } 0)\right)$$

- MRF incorporates prior knowledge that pixels that are spatially close are likely to be similar

x: segmentation
y: measured image/data
$\Pr(x|y) \sim \Pr(y|x) \cdot \Pr(x)$

Basic EMMPM Image Model

- The image that we desire to segment will be represented as the random vector **Y**. The value of the image at a particular pixel s will be represented as the random variable Y_s
- Each pixel is classified into a specific class, this collection of data is called the label field, denoted **X**.
- Class label at pixel s is represented as the random variable X_s.
- The number of distinct classes is represented by L, so $X_s \in \{1,2,\ldots,L\}$.
- Eight-point neighborhood system is used.

PURDUE ENGINEERING

- Label field follows the Ising model with an 8-point neighborhood, the probability function for X is:

Probability of classification of pixels

$$p_X(\mathbf{x}) = \frac{1}{z}\exp\left(-\sum_{\{r,s\}\in\mathcal{C}} \beta b(x_r, x_s) - \sum_{s\in S} \gamma_{x_s}\right)$$

where

$$b(x_r, x_s) = \begin{cases} 0 & \text{if } x_r = x_s \\ 1 & \text{if } x_r \neq x_s \end{cases}$$

- The spatial interaction parameter, β, is used to control the likelihood that two neighboring class labels will disagree.
- The γ parameters allow us to adjust the relative likelihoods of the various class labels.

- Assume that the random variables $Y_1, Y_2, \ldots, Y_N$ are conditionally independent given the pixel label field X.
- Assume that the conditional probability density function of Y_r given X depends only on the value of X at pixel location r.

$$
\begin{aligned}
f_{\mathrm{Y}|\mathrm{X}}(\mathrm{y}|\mathrm{x},\theta) &= \prod_{r=1}^{N} f_{Y_r|\mathrm{X}}(y_r|\mathrm{x},\theta) \\
&= \prod_{r=1}^{N} f_{Y_r|X_r}(y_r|x_r,\theta)
\end{aligned}
$$

PURDUE ENGINEERING

- Using Baye's rule to obtain the conditional probability mass function of X given Y, $p_{X|Y}(x|y, \theta)$. This will be needed to obtain the segmentation using EM/MPM.

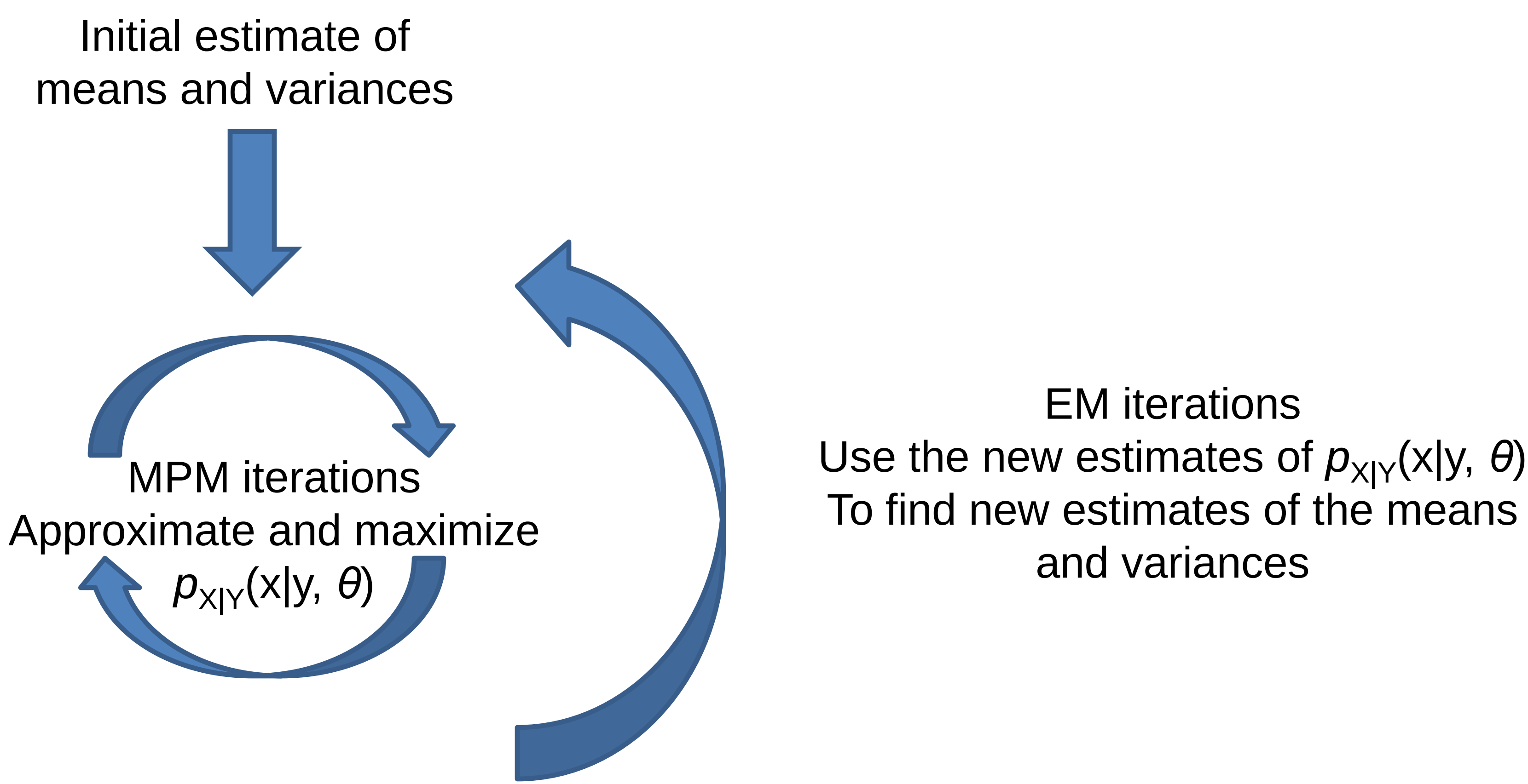
$$
\begin{aligned}
p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y},\theta) &= \frac{f_{\mathbf{Y}|\mathbf{X}}(\mathbf{y}|\mathbf{x},\theta)p_{\mathbf{X}}(\mathbf{x})}{f_{\mathbf{Y}}(\mathbf{y}|\theta)} \\
&= \frac{1}{f_{\mathbf{Y}}(\mathbf{y}|\theta)}\left[\zeta_{x,\theta}\exp\left(\xi_{x,y,\theta}\right)\right] \\
&\qquad \cdot \frac{1}{z}\exp\left(-\sum_{\{r,s\}\in C}\beta b(x_r,x_s)\right) \\
&= \frac{\zeta_{x,\theta}}{z f_{\mathbf{Y}}(\mathbf{y}|\theta)}\exp\left(\xi_{x,y,\theta}-\beta\delta_x\right)
\end{aligned}
$$

Probability of classification given the observed data

where

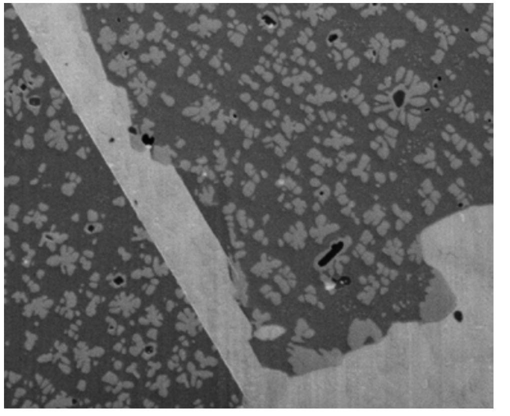
$$\delta_x = \sum_{\{r,s\}\in C} b(x_r,x_s)$$

EMMPM Algorithm



Results for Standard EMMPM

Beta 0.5

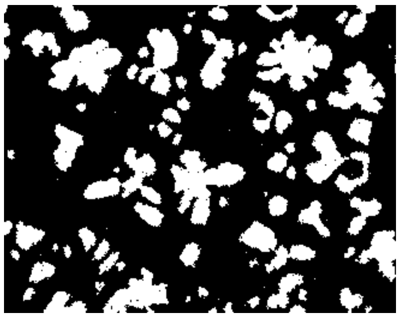


Original Image

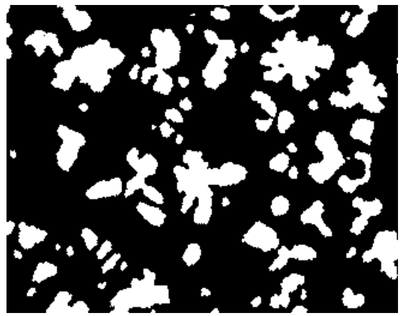
Beta 1

Beta 1.5

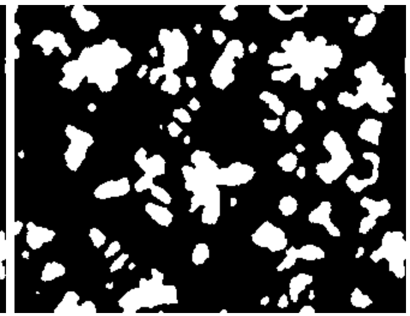
Results for Standard EMMPM



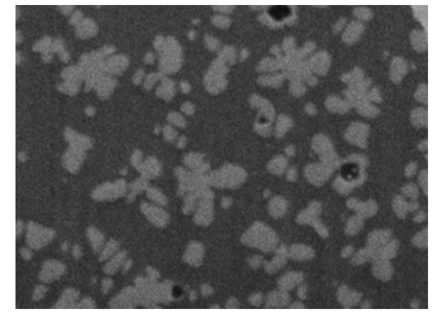
Beta 0.5



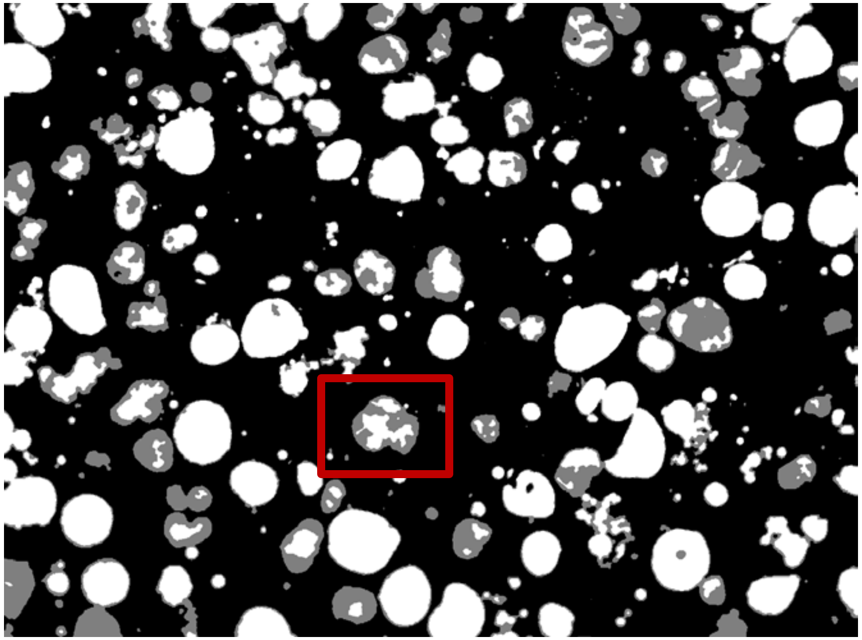
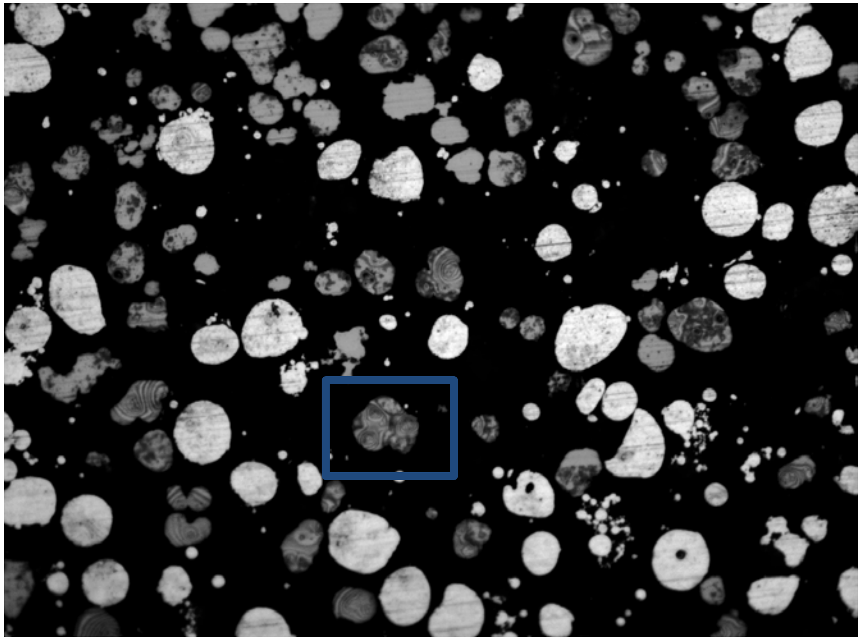
Beta 1



Beta 1.5



Original Image

Incorporating Edge Information

- MRF-based approach tends to be poor at precise boundary localization.
- Solution is to incorporate image gradient information into the segmentation process.
- This is done by directly manipulating the form of $p_{X|Y}(x|y, \theta)$.
- A new term is added and weighed by parameter β_e.

PURDUE ENGINEERING

Incorporating Edge Information

- Our aim is to cause the boundaries to be located at strong edges in the image.
- Penalties for two neighbors being classified differently should be reduced at locations where strong edges are present.
- Making the edge penalty term inversely proportional to the simple gradient at that boundary location.

$$
\begin{aligned}
e_p &= \sum_{\{r,s\} \in B_{\mathbf{x}}} \frac{\|r-s\|}{\|y_r - y_s\| + 0.5} \\
&= \sum_{\{r,s\} \in \mathcal{C}} \frac{\|r-s\|}{\|y_r - y_s\| + 0.5} b(x_r, x_s)
\end{aligned}
$$

PURDUE ENGINEERING

Incorporating Edge Information

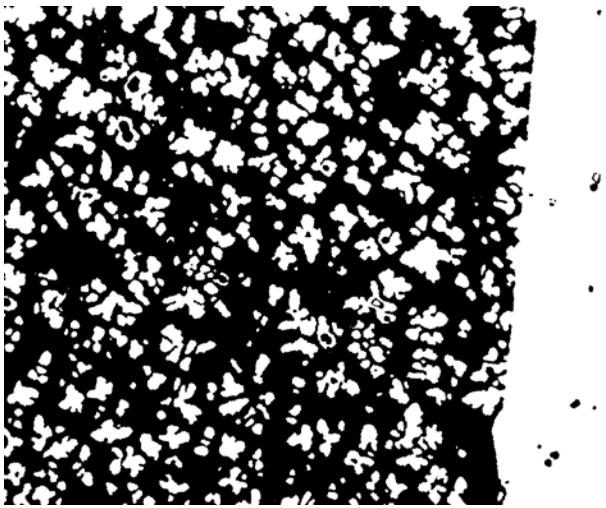
- The new edge penalty term is multiplied by β_e and added to the exponent of $p_{X|Y}(x|y, \theta)$.

$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y},\theta) = \frac{\zeta_{x,\theta}}{z'(\mathbf{y},\theta)} \exp\left(\xi_{x,y,\theta} - \beta\delta_x - \beta_e e_p\right)$$

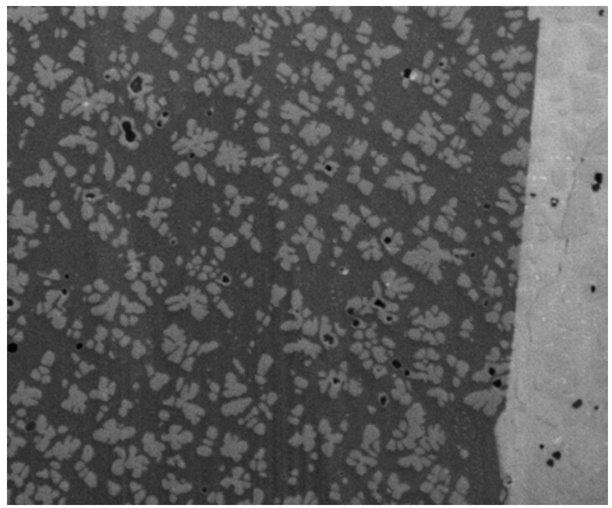
Spatial Interaction term

Gradient information term

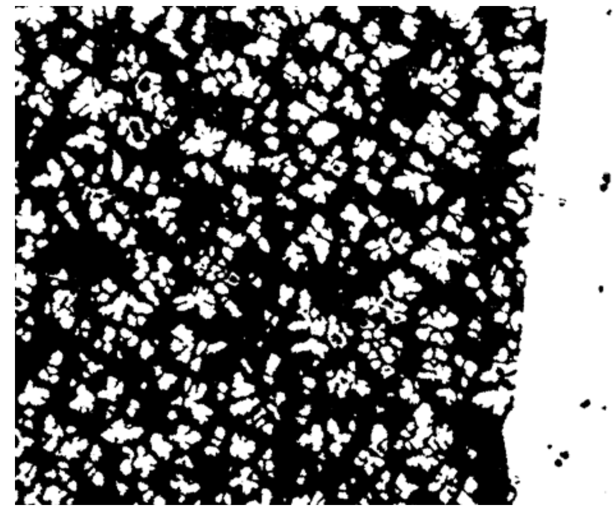
Results with edge information incorporated



Basic model, Beta = 1

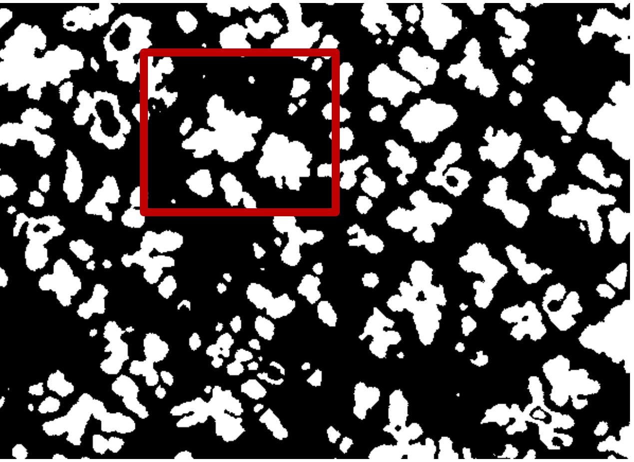


Original Image

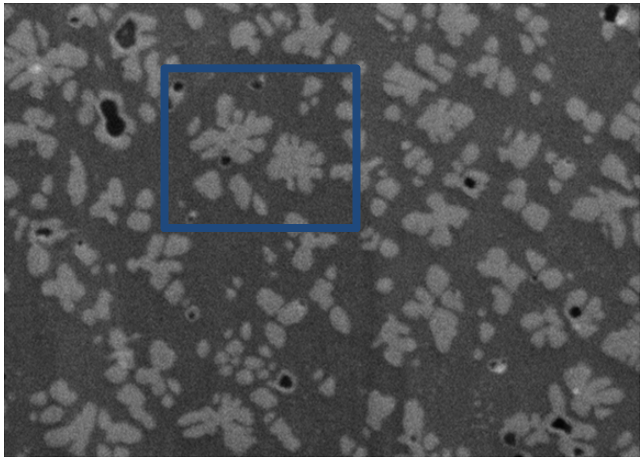


Beta =1, Beta E = 2

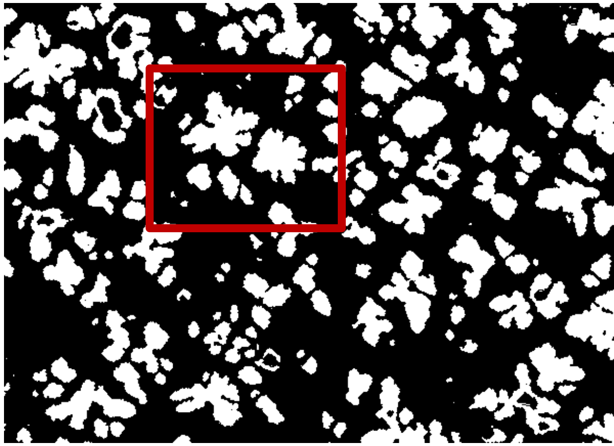
Results with edge information incorporated



Basic model, Beta = 1

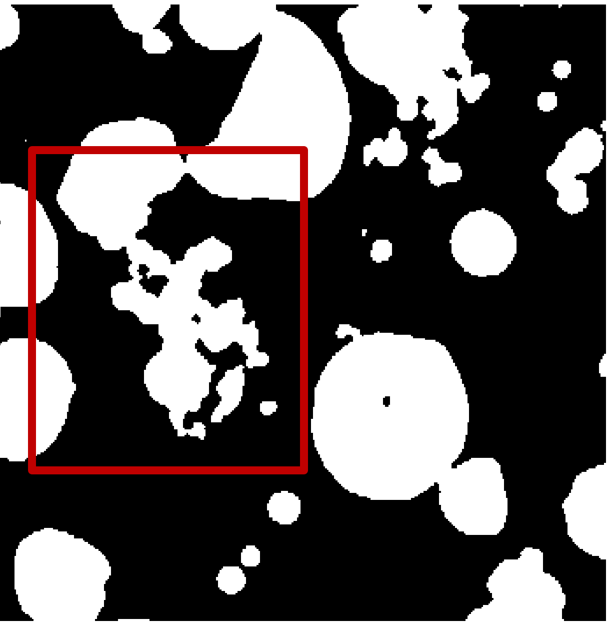


Original Image

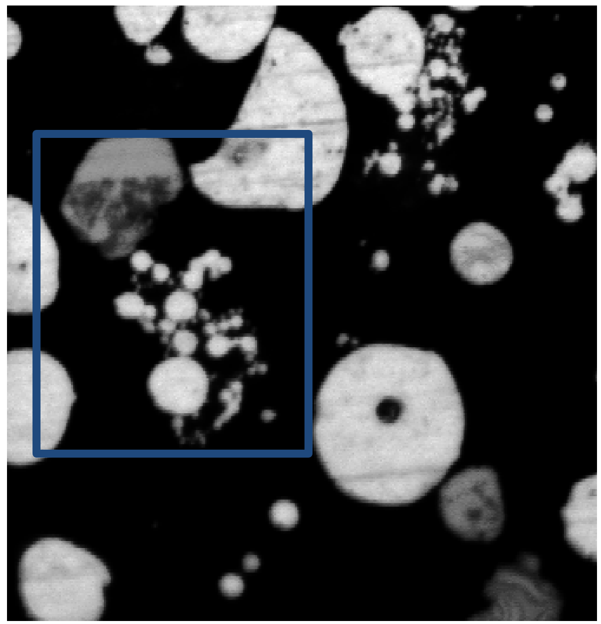


Beta =1, Beta E = 2

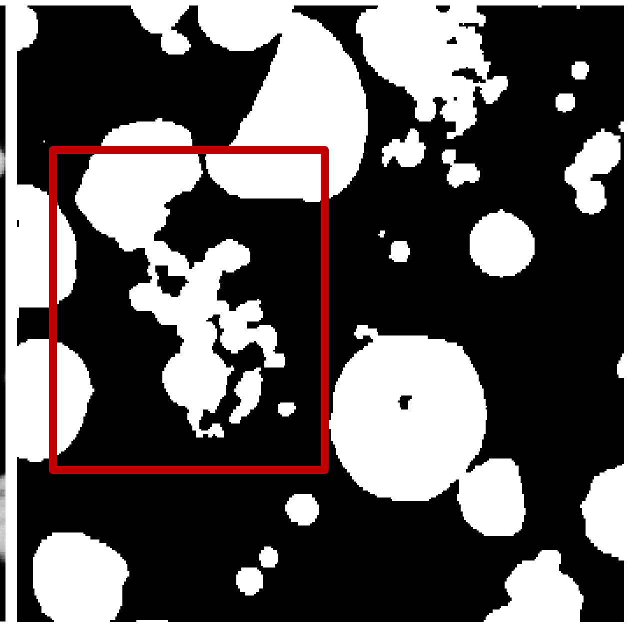
Results with edge information incorporated



Basic model, Beta = 1.5



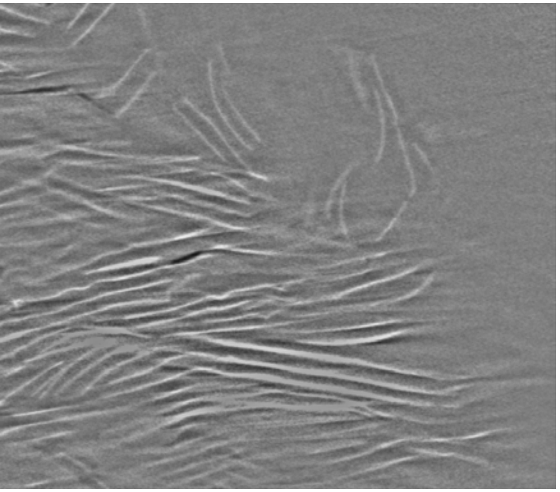
Original Image



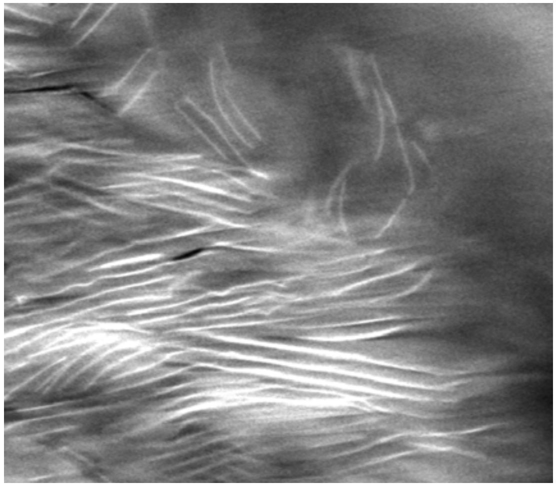
Beta =2, Beta E = 1.5

PURDUE
ENGINEERING

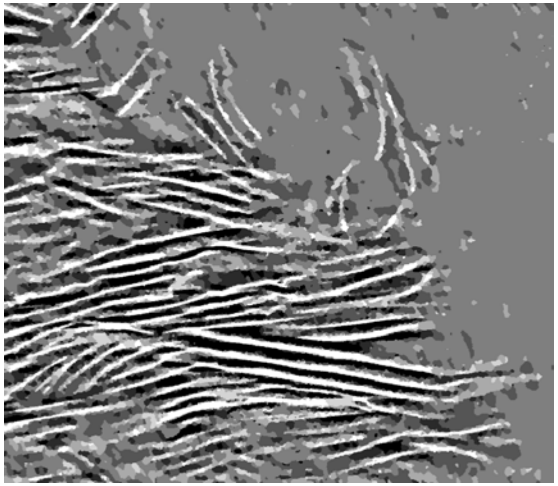
Preprocessing using high pass filter



High Pass filtered

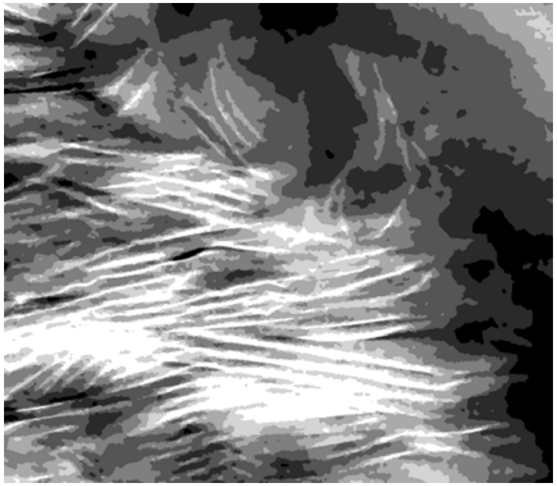


Original Image



Beta =1.5, Beta E =2

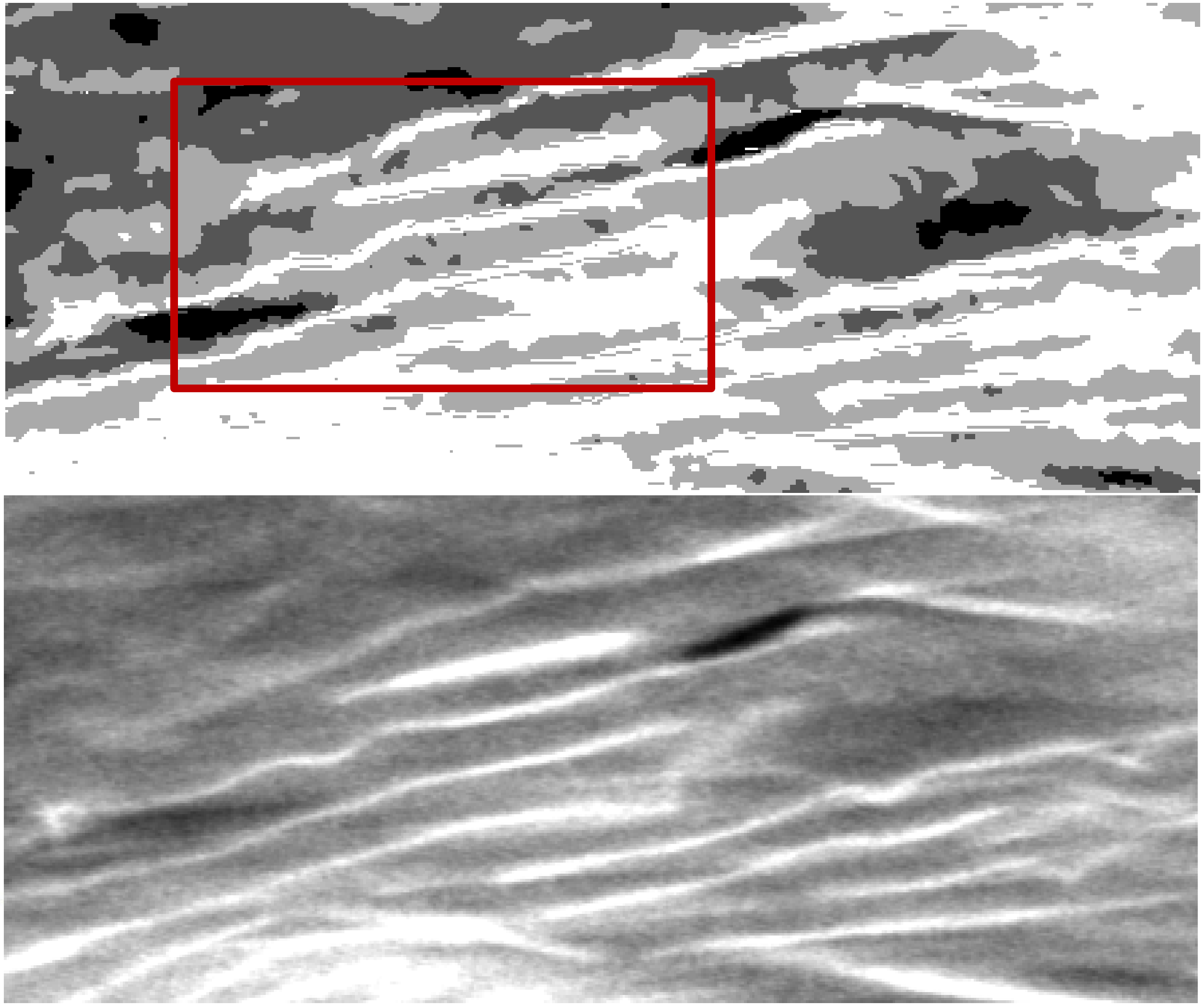
with preprocessing



Beta = 1.5, beta E = 2, no preprocessing

PURDUE
ENGINEERING

Wrong balance of Beta and Beta E



Incorporating curvature

- The point of considering curvature is to influence the shape of the resulting regions.
- We desire regions with smoother boundaries.
- To accomplish this we perform morphological filtering using more than one structuring element.
- Main point is how curved a boundary would have to be before a particular location could be included in a region.
- Use eight circular structuring elements with different radii.
- The maximum radius is specified by the user. (R max)

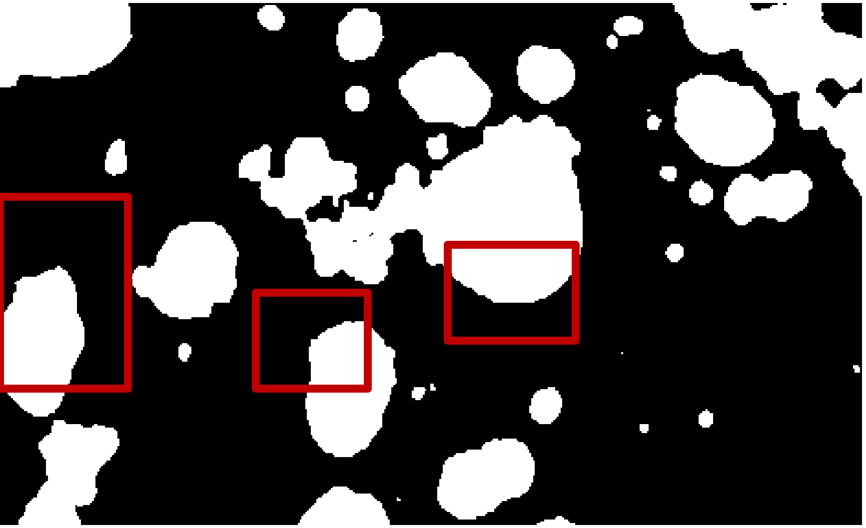
Incorporating curvature

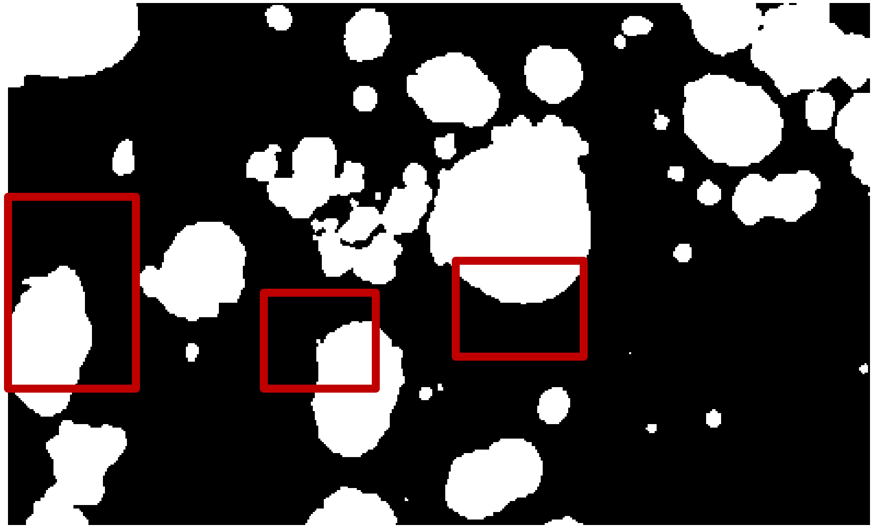
- View pixels which are excluded from a morphological opening of their class as misclassified.
- Add a penalty for assigning that pixel to that particular class in future iterations.
- Maintain a function fcrv
- Each time a pixel fails to be included in an opening of a specific class, we increment fcrv by a penalty proportional to the radius of the smallest structuring element under which it was excluded.

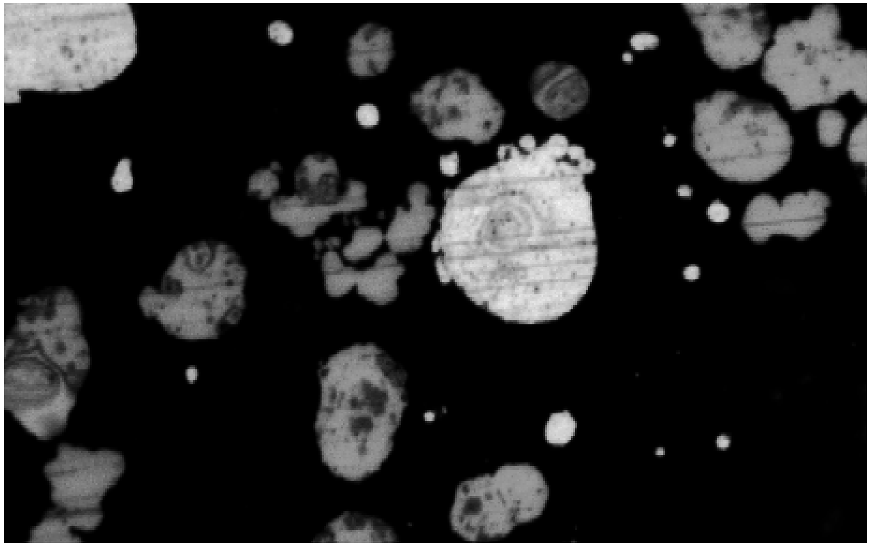
$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y},\theta) = \frac{\zeta_{x,\theta}}{z'(\mathbf{y},\theta)} \exp\left(\xi_{x,y,\theta} - \beta\delta_x - \beta_e e_p - \beta_c \sum_{s\in S} f_{crv}(s, x_s)\right)$$

Curvature penalty
weighed by β_c

Results for Incorporating curvature







PURDUE
ENGINEERING

Markov Chain Markov Random Field (MCMRF) model

- An extension of the previous 2D image model to using a Markov Chain to describe the 3D relationship
  - Extend the 2D prior model to a 3D model
  - Segmentation of a slice is correlated to the segmentation of previous slice.

MCMRF model—observation model

- Using Bayes' rule, we can obtain the posterior density as:

$$p_{\mathbf{X}|\mathbf{Y}}(\mathbf{x}|\mathbf{y},\theta) = \frac{\zeta_{x,\theta}}{z'(\mathbf{y},\theta)} \prod_{n=1}^{N} \exp\left(- \sum_{\{r,s\}\in C_n} \beta\, d(x(t_n,r),x(t_n,s)) - \sum_{s\in S} \gamma(x(t_{n-1},s'),x(t_n,s)) \right)$$

Spatial interaction within the neighborhood in the same slice

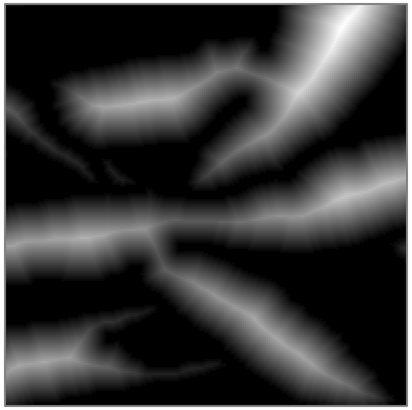
Penalty depends on the pixel label from the previous slice

probability of segmentation X=x given observed image Y=y

Segmentation with MCMRF using EM/MPM

- To use the MCMRF for segmentation of the serial-sectioned dataset, one important parameter to decide is the function γ
- Observation: the inner part of an object is more likely to stay unchanged from slice to slice
- Idea: use the Euclidean distance transform (EDT) to assign the function γ
  - Give higher cost to the inner part of objects to change its class label

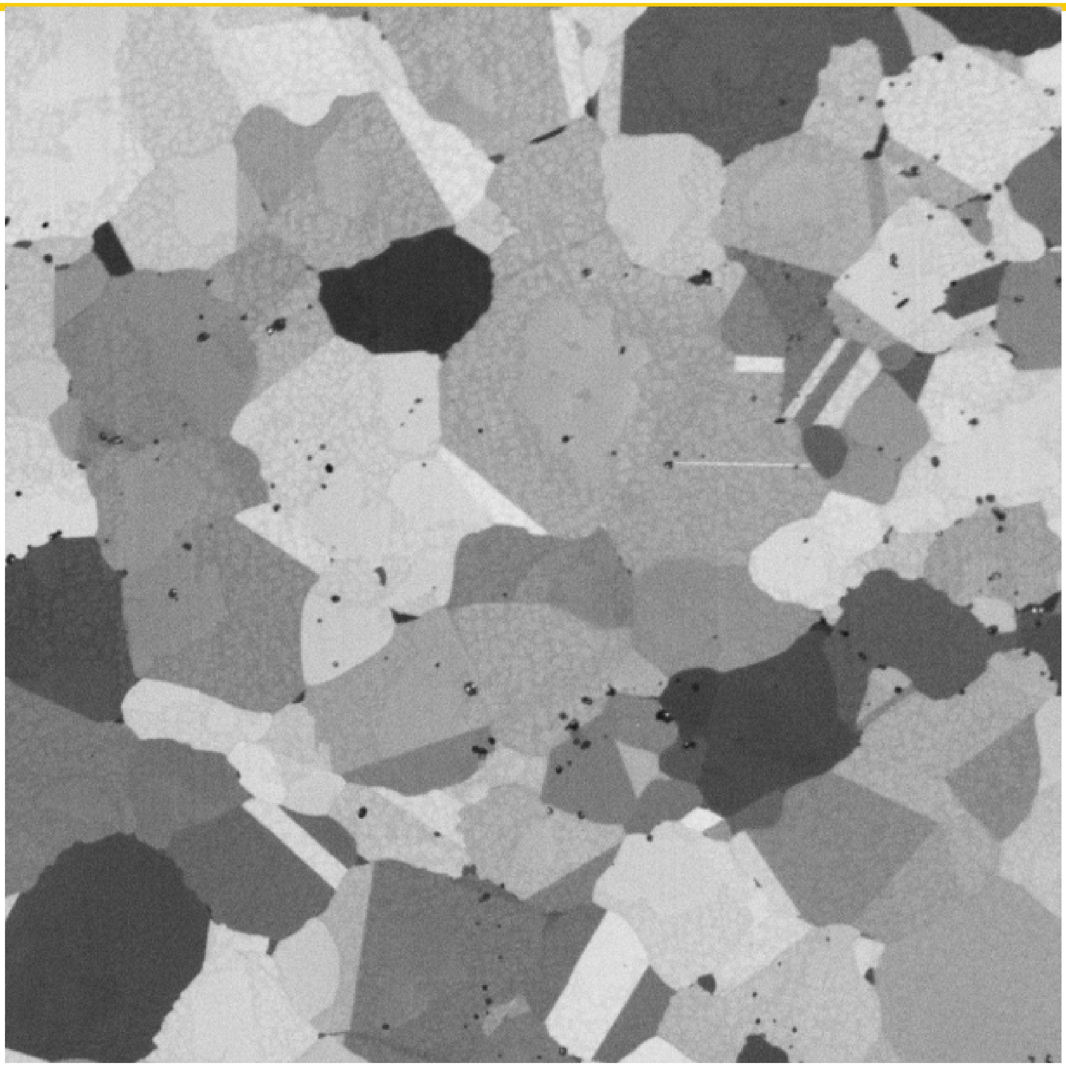
2D label field



EDT image

PURDUE ENGINEERING

IN-100

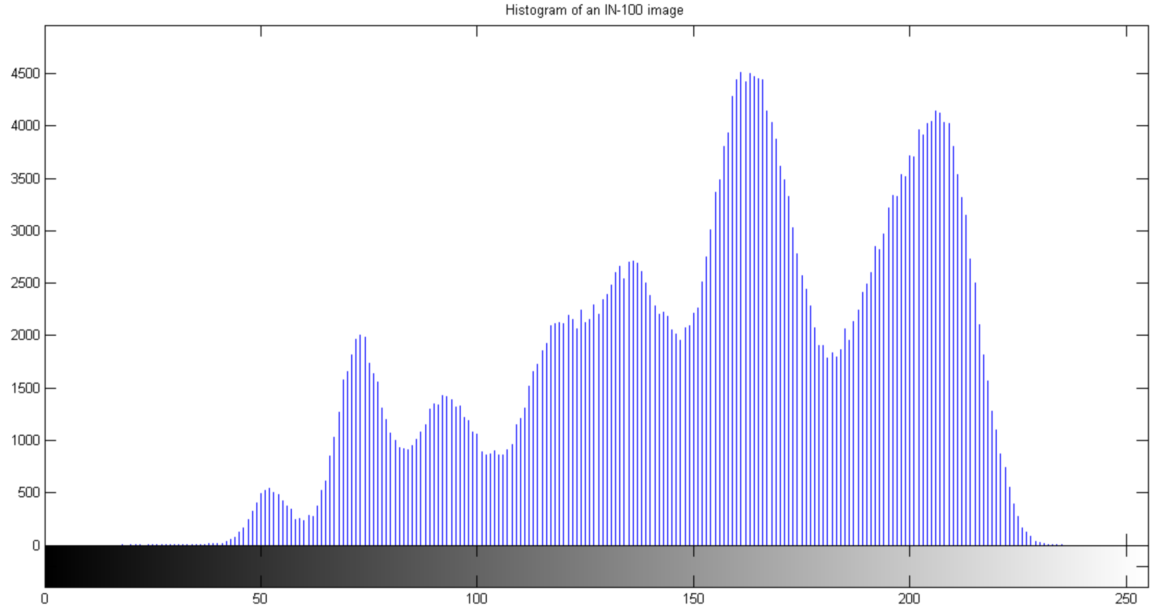


Segmentation of IN-100 dataset

- **Segmentation procedure:**

1. **Carbide removal**
2. **Segmentation with MCMRF using EM/MPM**

- Determining the number of classes using histogram:



PURDUE ENGINEERING

Heuristic carbide removal approach

- **Carbide detection**

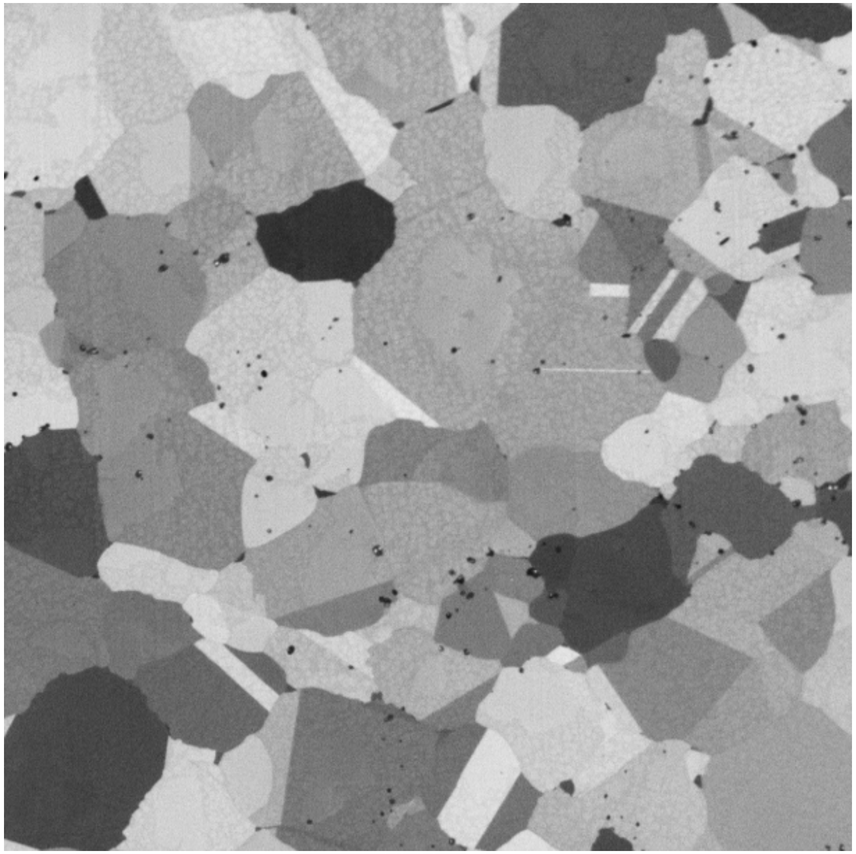
1. Apply a 3x3 local variance filter; retain small connected components; remove detected carbides
2. Threshold the last 10% pixels in grayscale values; retain small connected components; remove detected carbides
3. Apply the 2D EM/MPM algorithm to identify the remaining carbides as the pixels classified with labels of zeros; retain small connected components; remove detected carbides

- **Carbide removal**

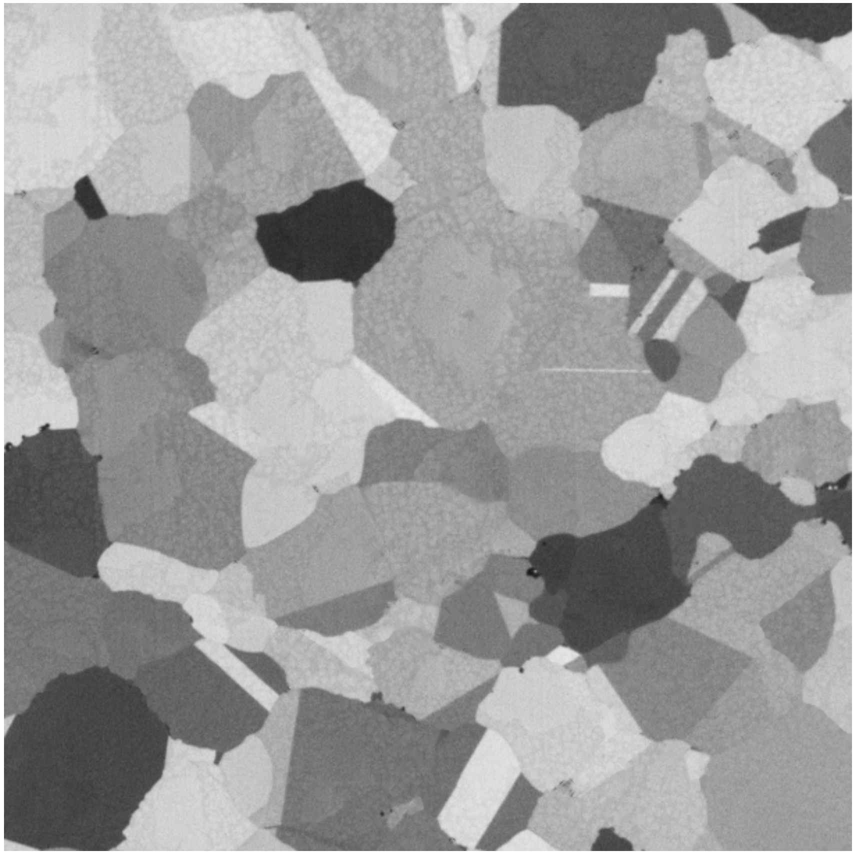
1. Determine the boundary carbide pixels
2. Using a 5x5 local average filter to replace carbide pixels with the averaged value from neighboring non-carbide pixels, and then mark those pixels as non-carbide ones
3. Repeat step 1-2 until there is no carbide pixels in an image

PURDUE ENGINEERING

Carbide-removed images



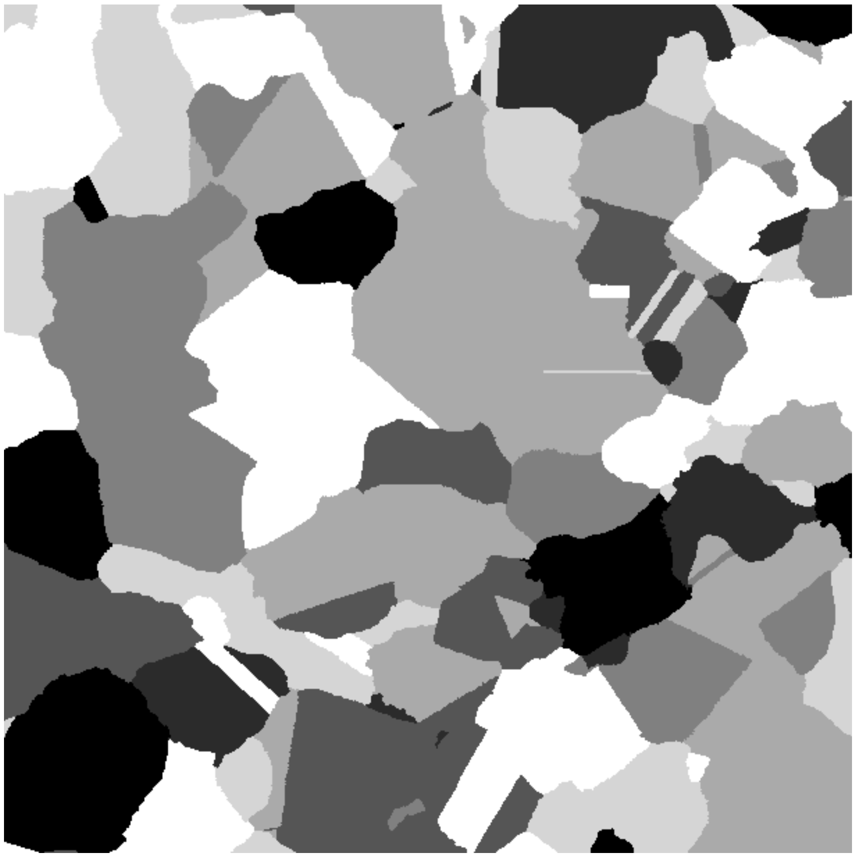
original image



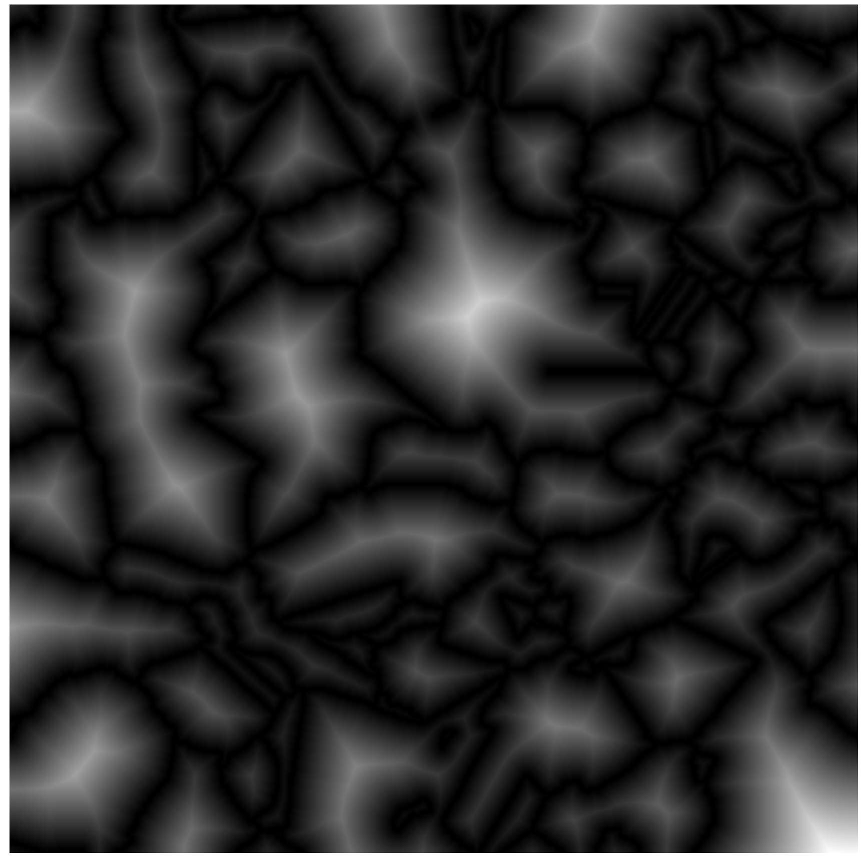
carbide-removed image

PURDUE
ENGINEERING

Modified EDT for assigning function γ

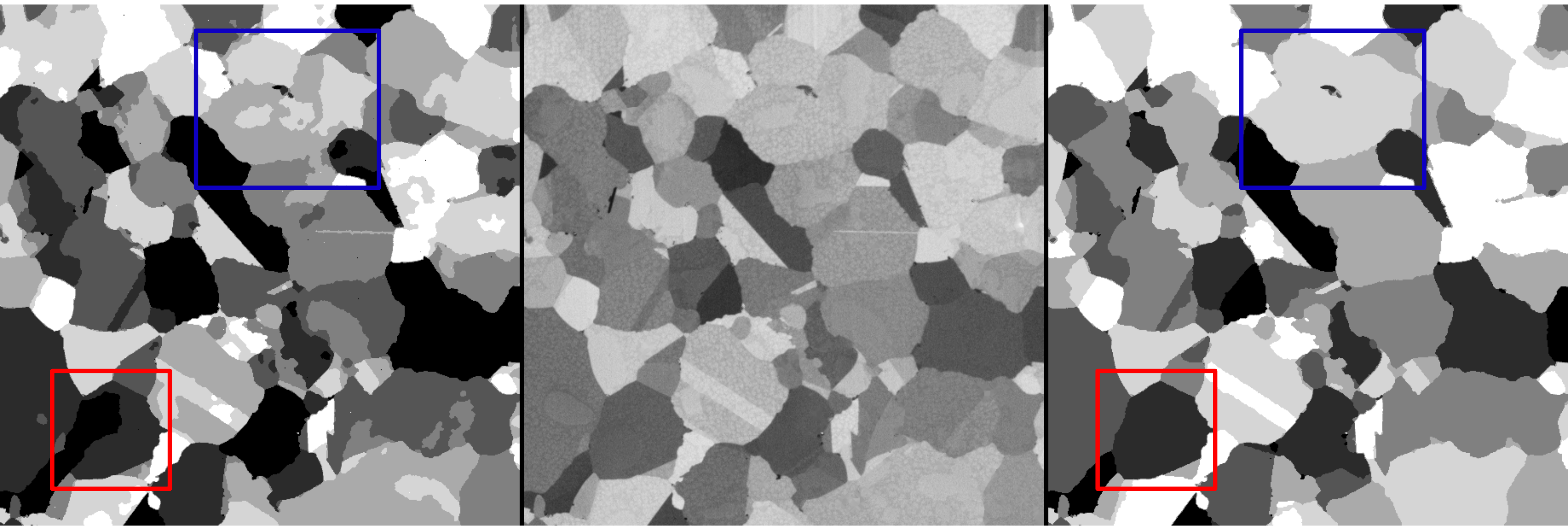


Label Field



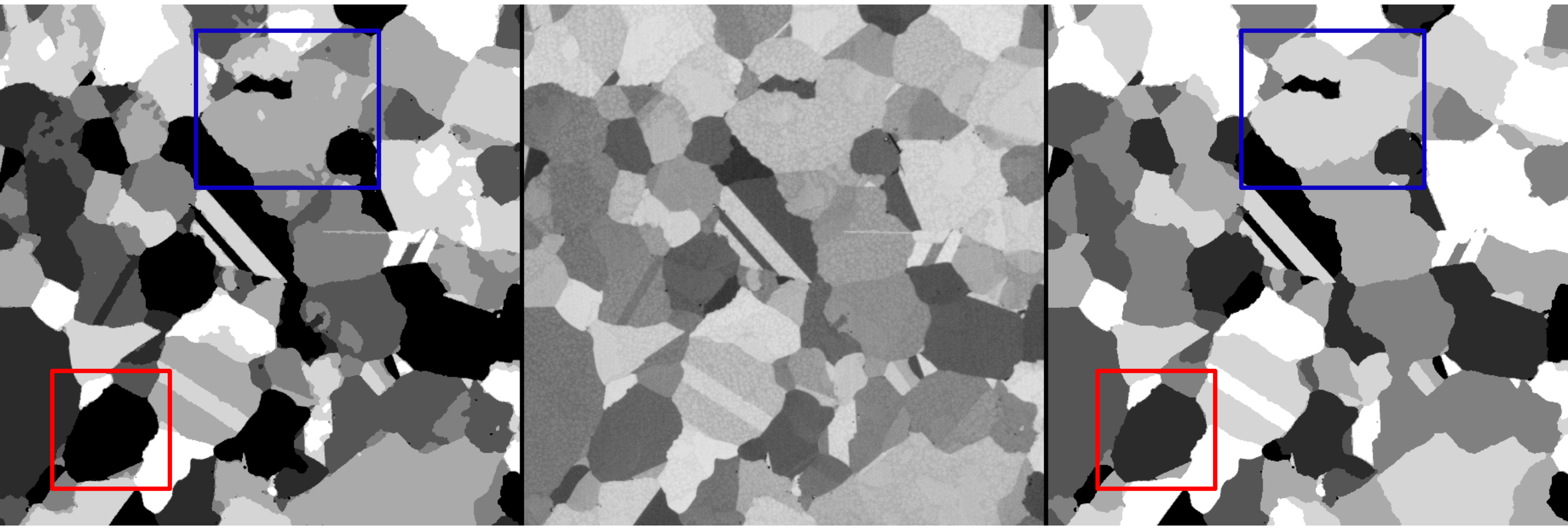
Modified EDT

Comparison with MRF prior



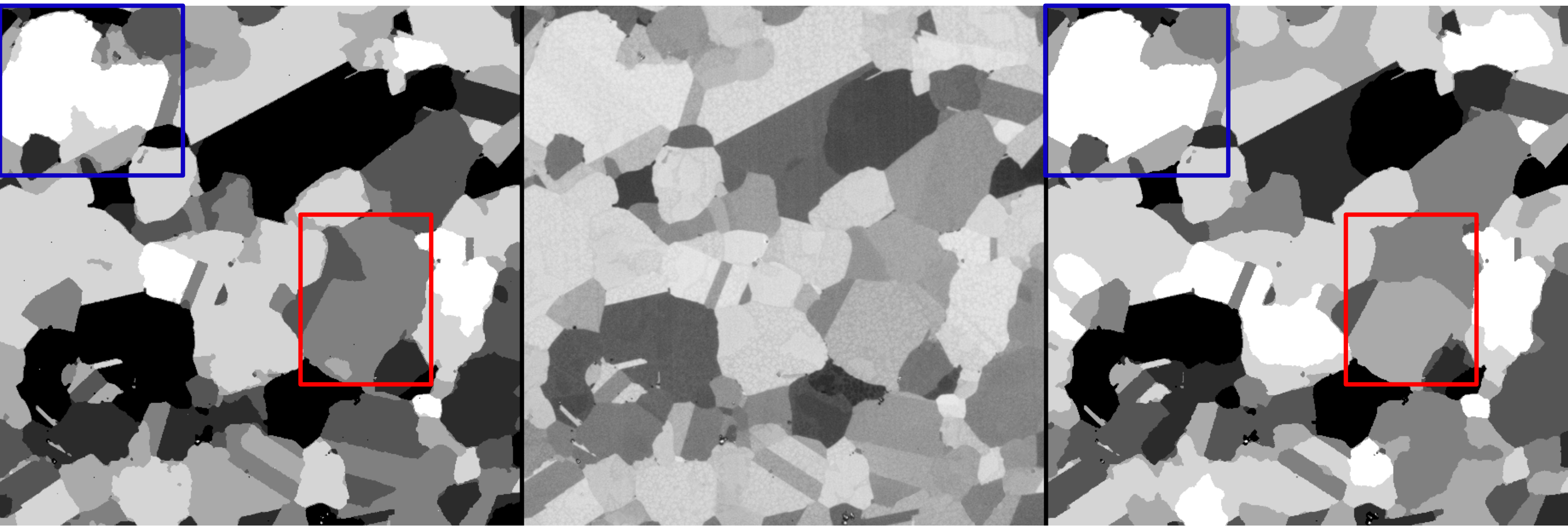
2D MRF　　　original　　　MCMRF

Comparison with MRF prior



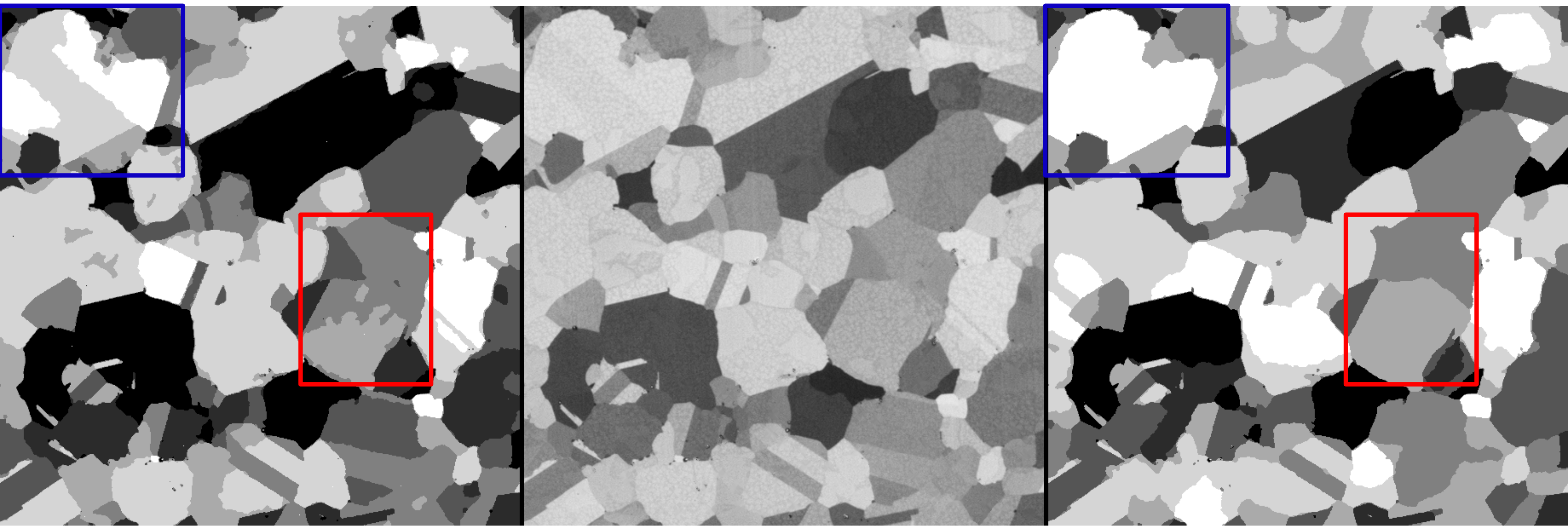
2D MRF   original   MCMRF

Comparison with MRF prior



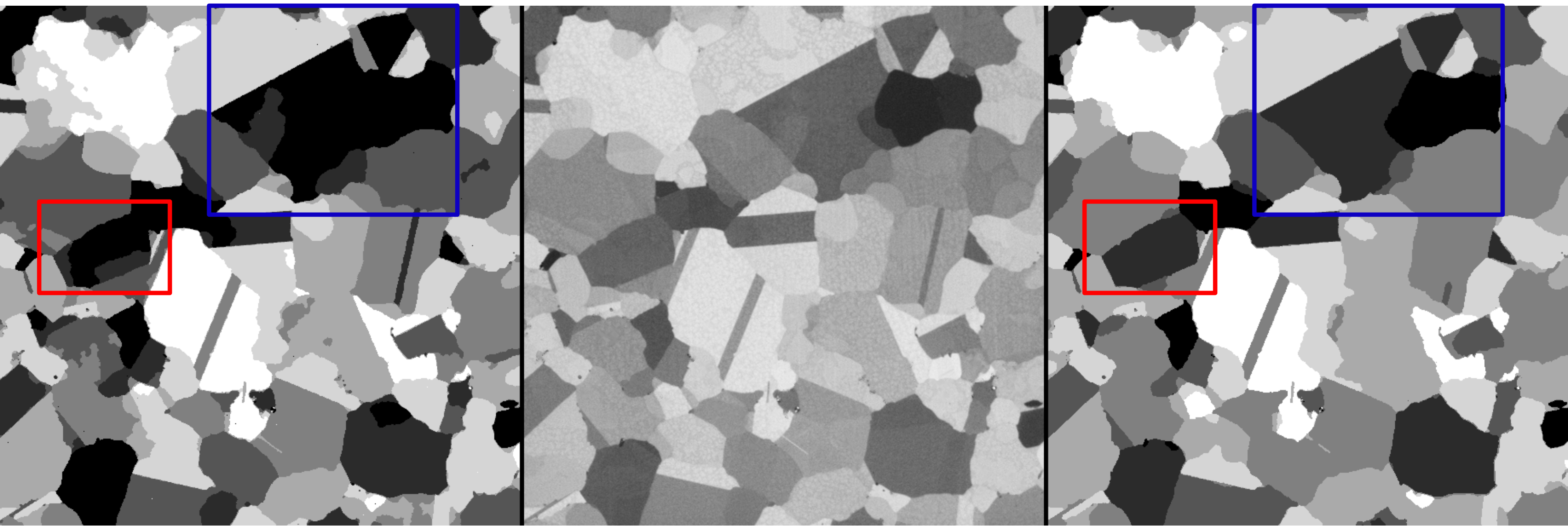
2D MRF original MCMRF

Comparison with MRF prior



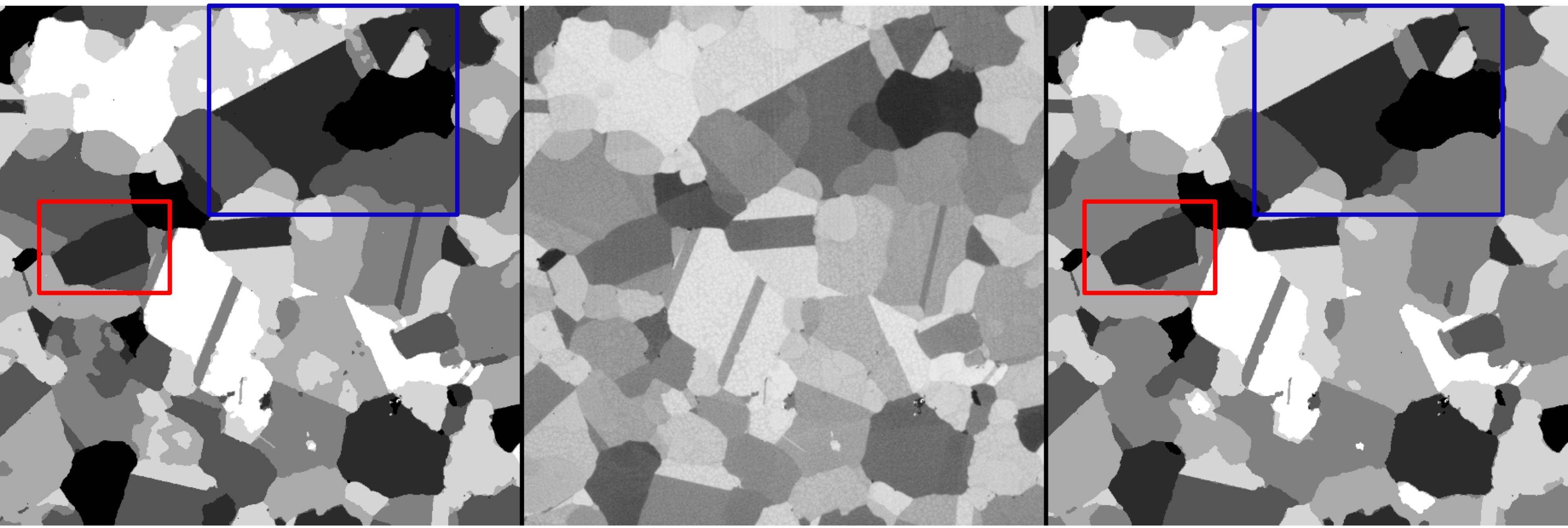
2D MRF | original | MCMRF

Comparison with MRF prior



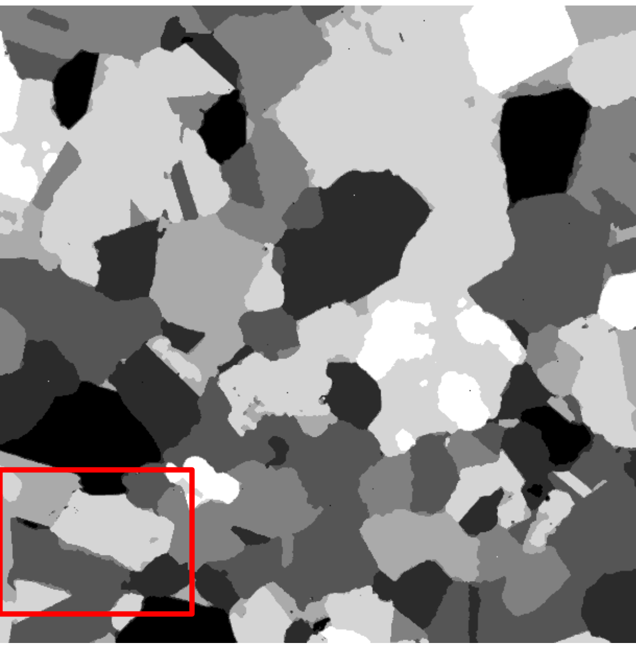
2D MRF original MCMRF

Comparison with MRF prior

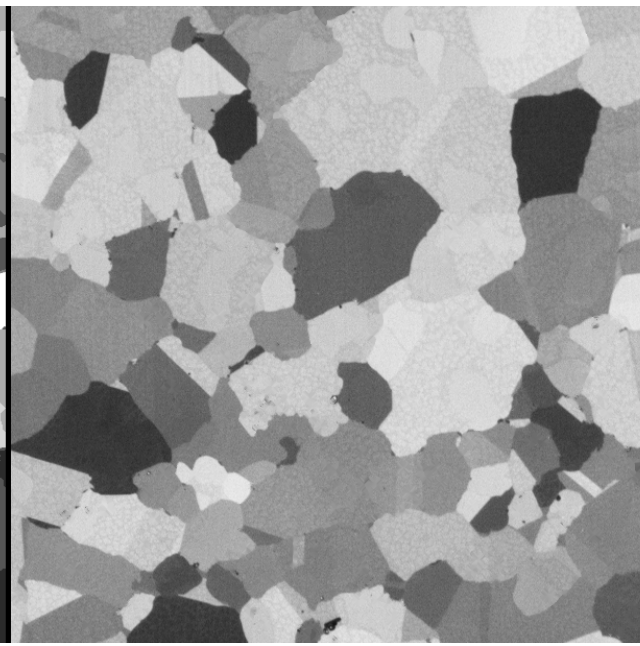


2D MRF original MCMRF

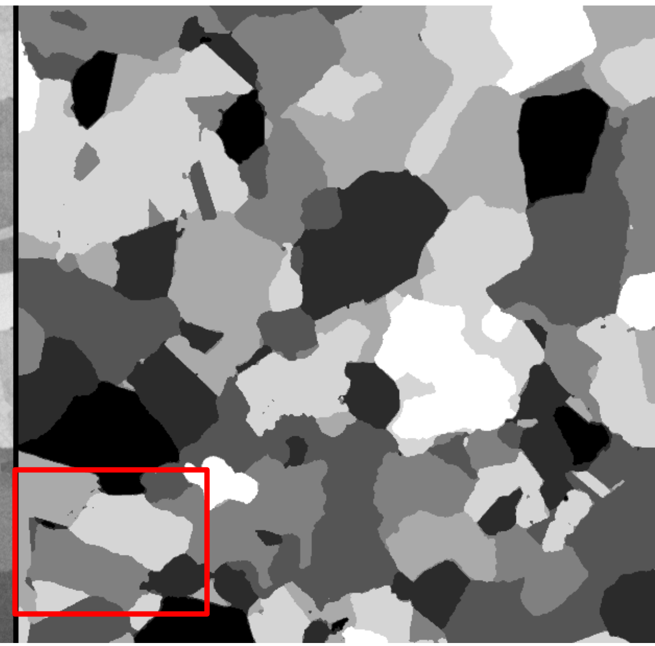
Comparison with MRF prior



2D MRF

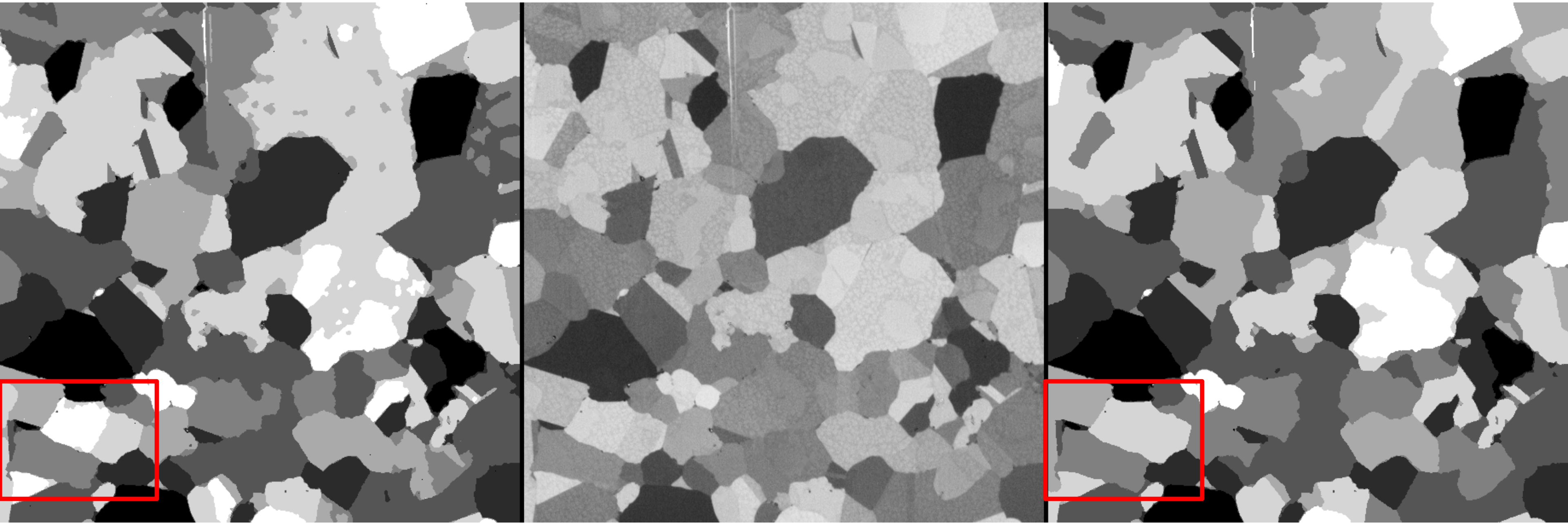


original



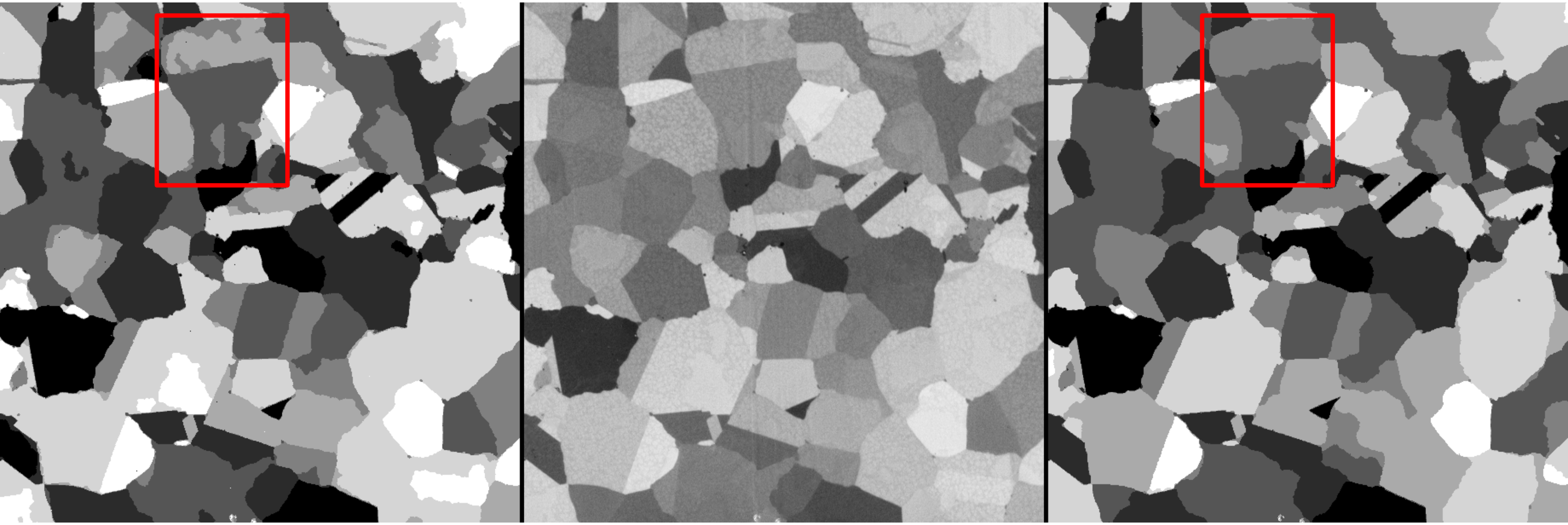
MCMRF

Comparison with MRF prior



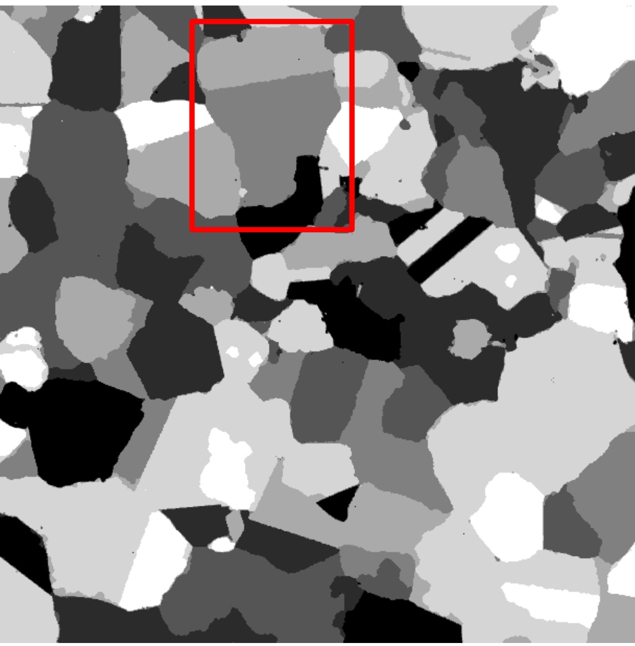
2D MRF | original | MCMRF

Comparison with MRF prior

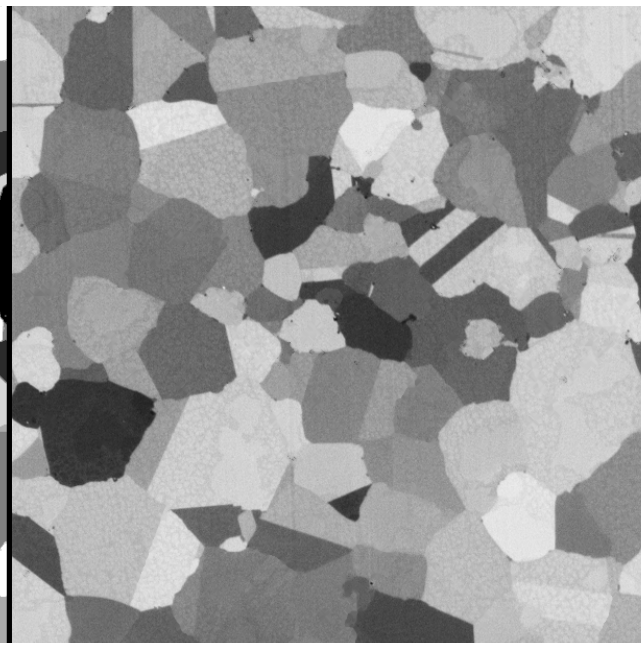


2D MRF original MCMRF

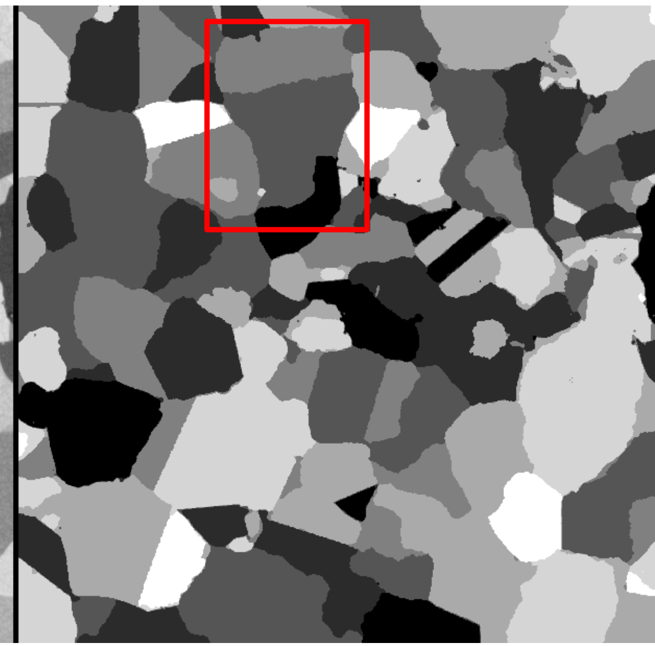
Comparison with MRF prior



2D MRF



original



MCMRF