Algorithm 1 Proposed Noise Suppression Method

Require: Multi-energy CT images $\vec{\boldsymbol{I}} = (\boldsymbol{I}_1, \ldots, \boldsymbol{I}_m)$, Decomposition matrix M, Hyper-parameters (η, a, λ, D, X).

$\vec{\sigma} \leftarrow$ Noise standard deviations of $\vec{\boldsymbol{I}}$
$A \leftarrow (M^{\mathrm{T}}M)^{-1}M^{\mathrm{T}}\mathrm{diag}(\vec{\sigma})$
$U, \Sigma, V^{\mathrm{T}} \leftarrow \mathrm{SVD}(A)$ // Entries of Σ placed in non-descending order.
// Perform basis transformation for each pixel.
for pixel at $\vec{x}$ in $\vec{\boldsymbol{I}}$ **do**
  $\vec{\mu}(\vec{x}) \leftarrow (\boldsymbol{I}_1(\vec{x}), \ldots, \boldsymbol{I}_m(\vec{x}))$
  $\vec{a}(\vec{x}) \leftarrow (M^{\mathrm{T}}M)^{-1}M^{\mathrm{T}}\vec{\mu}(\vec{x})$
  $\vec{b}(\vec{x}) \leftarrow U^{\mathrm{T}}\vec{a}(\vec{x})$
  // Reformat $\vec{b}$ to intermediate images $\vec{\boldsymbol{I}}_b$.
  $(\boldsymbol{I}_{b1}(\vec{x}), \ldots, \boldsymbol{I}_{bn}(\vec{x})) \leftarrow \vec{b}(\vec{x})$
end for
// Perform selective filtering for each pixel.
$\sigma_{b1} \leftarrow$ Noise standard deviation of $\boldsymbol{I}_{b1}$
for pixel at $\vec{x}$ in $\vec{\boldsymbol{I}}$ **do**
  for pixel at $\vec{y}$ in neighbourhood of $\vec{x}$ **do**
    $w_d(\vec{x}, \vec{y}) \leftarrow \exp\left[-\frac{\|\vec{x}-\vec{y}\|_2^2}{(\eta a)^2}\right] \times \mathrm{Th}(|\vec{x} - \vec{y}| \leq D)$
    $w_{\boldsymbol{I}_{b1}}(\vec{x}, \vec{y}) \leftarrow \exp\left[-\frac{\|\boldsymbol{I}_{b1}(\vec{x})-\boldsymbol{I}_{b1}(\vec{y})\|_2^2}{(\lambda\sigma_{b1})^2}\right] \times \mathrm{Th}(|\boldsymbol{I}_{b1}(\vec{x}) - \boldsymbol{I}_{b1}(\vec{y})| \leq X)$
    $w(\vec{x}, \vec{y}) \leftarrow w_d(\vec{x}, \vec{y}) w_{\boldsymbol{I}_{b1}}(\vec{x}, \vec{y})$
  end for
  for $i = 2 : m$ **do**
    $\boldsymbol{I}'_{bi}(\vec{x}) \leftarrow \frac{\sum_{\vec{y}} \boldsymbol{I}_{bi}(\vec{y}) w(\vec{x}, \vec{y})}{\sum_{\vec{y}} w(\vec{x}, \vec{y})}$
  end for
end for
// Transform $\vec{b}'$ to denoised $\vec{a}'$ for each pixel.
for pixel at $\vec{x}$ in $\vec{\boldsymbol{I}}$ **do**
  $\vec{b}'(\vec{x}) \leftarrow (\boldsymbol{I}_{b1}(\vec{x}), \boldsymbol{I}'_{b2}(\vec{x}) \ldots, \boldsymbol{I}'_{bn}(\vec{x}))$
  $\vec{a}'(\vec{x}) \leftarrow U\vec{b}'(\vec{x})$
  // Reformat $\vec{a}'$ to denoised material basis images $\vec{\boldsymbol{I}}'_a$.
  $(\boldsymbol{I}'_{a1}(\vec{x}), \ldots, \boldsymbol{I}'_{an}(\vec{x})) \leftarrow \vec{a}'(\vec{x})$
end for
return $\vec{\boldsymbol{I}}'_a$