REMOVING OF COMBINED NOISE IN RASTER IMAGES

DVOENKO S.D.[1] , DANG N.H. THANH[2]

[1]TULA STATE UNIVERSITY, TULA, RUSSIAN FEDERATION

[2]HUE INDUSTRIAL COLLEGE, HUE, VIETNAM

SPEAKER: DVOENKO S.D.

BARCELONA 10/2016

47 slides

1. DENOISING PROBLEM

Today imaging science has an important development and has many applications in different fields of life. The researched object of imaging science is digital image that can be created by many digital devices. One of the limits of using digital devices to create digital images is noise. Noise reduces the image quality. It appears in almost types of images, including biomedical and electronic microscopy images. The type of noise in this case can be considered as combination of Gaussian and Poisson noises.

In this paper we propose a method to remove noise by using total variation. Our method is developed with the goal to combine two famous models: ROF for removing Gaussian noise and modified ROF for removing Poisson noise. As a result, our proposed method can be also applied to remove Gaussian or Poisson noise separately. The proposed method can be applied with automatically evaluated parameters (unknown noise for real images).

1. DENOISING PROBLEM

Let in $\mathbb{R}^2$ space, a bounded domain $\Omega \in \mathbb{R}^2$ be given. We call function $u(x, y) \in \mathbb{R}^2, v(x, y) \in \mathbb{R}^2$ respectively ideal image (without noise) and observed image (noisy image), where $\Omega \in \mathbb{R}^2$.

In practice, v is given and we have to find u. This is denoising problem.

If function u is smooth then its total variation is defined by:

$$V_T[u] = \int_{\Omega} |\nabla u| dxdy \,,$$

where $\nabla u = (u_x, u_y), u_x = \frac{\partial u}{\partial x}, u_y = \frac{\partial u}{\partial y}, |\nabla u| = \sqrt{u_x^2 + u_y^2}.$

We consider in this case, total variation is always bounded.

1. DENOISING PROBLEM

- The problem of noise removal is actual today and there are many different strong approaches to solve it.
- Variational approach is one of them and it's pioneered by Rudin.
- Rudin proposed the denoising model, that was called ROF:

$$\begin{cases} u^* = \arg\min\limits_u \int_\Omega |\nabla u| dxdy \\ \int_\Omega (u - v)^2 dxdy = \sigma^2, \end{cases}$$

where σ – variance of Gaussian noise. Another simpler form:

$$u^* = \arg\min_u \int_\Omega |\nabla u| dxdy + \frac{\lambda}{2} \int_\Omega (u - v)^2 dxdy,$$

1. DENOISING PROBLEM

$$u^* = \arg\min_u \int_\Omega |\nabla u| dxdy + \frac{\lambda}{2}\int_\Omega (u - v)^2 dxdy,$$

where λ – Lagrange multiplier.

- The first term – regularization term. The second term – data fidelity term.
- ROF model is only designed to remove Gaussian noise.
- In order to remove Poisson noise, Le T. developed the modified ROF model:

$$u^* = \arg\min_u \int_\Omega |\nabla u| dxdy + \beta \int_\Omega \big(u - v\ln(u)\big) dxdy,$$

where β – regularization coefficient.

1. DENOISING PROBLEM

- Gaussian noise is popular and it always appears in digital image. Poisson noise, for example, is a result of X-ray devices in medicine.
- Another type of noise – the combination of Gaussian and Poisson noises is also important. This type of noise appears, for example, in biomedical images (electronic microscopy images).
- **Can we use ROF or modified ROF to treat this combination of noise? (Yes, but ineffectively)**
- Today, we design a new model to treat this combination of noises more effectively by considering the proportion between Gaussian and Poisson noises.

2. MODEL TO REMOVE POISSON-GAUSSIAN NOISE

- We have to notice that: the total variation of noisy image is always greater than total variation of smoothed image. So if we want to denoise, we can use this characteristic:

$$V_T[u] \rightarrow min.$$

- We need to add a constraint to above optimization problem. We assume that with given image u, the mixed noise in image is fixed too (because Poisson noise is unchangeable and Gaussian noise only depends on noise variance):

$$\int_\Omega \ln\big(p(v|u)\big)\, dxdy = const, \qquad (1)$$

where $p(v|u)$ is conditional probability.

2. MODEL TO REMOVE POISSON-GAUSSIAN NOISE

- We consider the Gaussian noise. Its probability density function is:

$$p_1(v|u) = \frac{\exp\left(-\frac{(v-u)^2}{2\sigma^2}\right)}{\sigma\sqrt{2\pi}}.$$

- Analogously, we consider the Poisson noise. Its distribution function is:

$$p_2(v|u) = \frac{\exp(-u)u^v}{v!},$$

we have to notice that intensity levels of image colour are integer (for example, 8-bit grayscale image), so we regard u as an integer value, but this will ultimately be unnecessary.

2. MODEL TO REMOVE POISSON-GAUSSIAN NOISE

In order to treat the combination of Gaussian and Poisson noise, we consider the following linear combination:

$$\ln\big(p(v|u)\big) = \lambda_1 \ln\left(p_1(v|u)\right) + \lambda_2 \ln\left(p_2(v|u)\right),$$

where $\lambda_1 > 0, \lambda_2 > 0, \lambda_1 + \lambda_2 = 1.$

So from (1) we obtain the following constraint:

$$\int_\Omega \left(\frac{\lambda_1}{2\sigma^2}(v-u)^2 + \lambda_2\big(u - vln(u)\big)\right) dxdy = \kappa,$$

where κ – constant value.

(1) $\int\limits_\Omega \ln(p(v|u))dxdy = \text{const}$

2. MODEL TO REMOVE POISSON-GAUSSIAN NOISE

Mixed noise removal problem can be displayed in following form:

$$\begin{cases} u^* = \arg\min_u \int_\Omega |\nabla u| dxdy \\ \int_\Omega \left(\frac{\lambda_1}{2\sigma^2} (v-u)^2 + \lambda_2 \big(u - v ln(u)\big) \right) dxdy = \kappa. \end{cases}$$

We can transform this constrained optimization problem to the unconstrained optimization problem by using Lagrange functional:

$$L(u,\tau) = \int_\Omega |\nabla u| dxdy + \tau \left(\frac{\lambda_1}{2\sigma^2} \int_\Omega (v-u)^2 dxdy + \lambda_2 \int_\Omega \big(u - v ln(u)\big) dxdy - \kappa \right),$$

to find

$$(u^*, \tau^*) = \arg\min_{u,\tau} L(u,\tau)\,, \text{where } \tau > 0 \ - \text{Lagrange multiplier.}$$

2. MODEL TO REMOVE POISSON-GAUSSIAN NOISE

$$L(u,\tau) = \int_\Omega |\nabla u| dxdy + \tau \left(\int_\Omega \left(\frac{\lambda_1}{2\sigma^2}(v-u)^2 + \lambda_2 \big(u - v ln(u)\big) \right) dxdy - \kappa \right),$$

to find

$$(u^*, \tau^*) = \arg\min_u L(u,\tau)\,, \text{where } \tau \ - \text{Lagrange multiplier.} \qquad (2)$$

- If $\lambda_1 = 0, \beta = \lambda_2\tau$, our proposed model become modified ROF model to treat Poisson noise.
- If $\lambda_2 = 0, \lambda = \lambda_1/(2\sigma^2)$, our proposed model become ROF model to treat Gaussian noise.
- If $\lambda_1 > 0$ and $\lambda_2 > 0$, our proposed model is used to treat mixed Poisson-Gaussian noise.

3. MODEL DISCRETIZATION

Let function $f(x, y)$ be defined in bounded domain $\Omega \in \mathbb{R}^2$ and be the second-order continuous differentiable one by x and y for $(x, y) \in \Omega$.

We consider the special convex functional $F(x, y, f, f_x, f_y)$, where $f_x = \frac{\partial f}{\partial x}, f_y = \frac{\partial f}{\partial y}$. The solution of the optimization problem:

$$\int_\Omega F(x, y, f, f_x, f_y)dxdy \to min$$

satisfies the following Euler-Lagrange equation

$$F_f(x, y, f, f_x, f_y) - \frac{\partial}{\partial x}F_{f_x}(x, y, f, f_x, f_y) - \frac{\partial}{\partial y}F_{f_y}(x, y, f, f_x, f_y) = 0,$$

where $F_f = \frac{\partial F}{\partial f}, F_{f_x} = \frac{\partial F}{\partial f_x}, F_{f_y} = \frac{\partial F}{\partial f_y}$.

3. MODEL DISCRETIZATION

The Euler-Lagrange equation of problem (2) to find (u^*, τ^*) is:

$$-\frac{\lambda_1}{\sigma^2}(v-u)+\lambda_2\left(1-\frac{v}{u}\right)-\mu\frac{\partial}{\partial x}\left(\frac{u_x}{\sqrt{u_x^2+u_y^2}}\right)-\mu\frac{\partial}{\partial y}\left(\frac{u_y}{\sqrt{u_x^2+u_y^2}}\right)=0, \qquad (3)$$

where $\mu = 1/\tau$.

We can simplify this equation (3) to

$$\frac{\lambda_1}{\sigma^2}(v-u)-\lambda_2\left(1-\frac{v}{u}\right)+\mu\frac{u_{xx}u_y^2-2u_xu_yu_{xy}+u_x^2u_{yy}}{(u_x^2+u_y^2)^{\frac{3}{2}}}=0, \qquad (4)$$

where $u_{xx}=\frac{\partial^2 u}{\partial x^2}, u_{yy}=\frac{\partial^2 u}{\partial y^2}, u_{xy}=\frac{\partial}{\partial x}\left(\frac{\partial u}{\partial y}\right)=\frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x}\right)=u_{yx}$.

3. MODEL DISCRETIZATION

$$\frac{\lambda_1}{\sigma^2}(v-u)-\lambda_2\left(1-\frac{v}{u}\right)+\mu\frac{u_{xx}u_y^2-2u_xu_yu_{xy}+u_x^2u_{yy}}{\left(u_x^2+u_y^2\right)^{\frac{3}{2}}}=0. \qquad (4)$$

In order to discretize the equation (4), we add an artificial time parameter and consider the function $u=u(x,y,t)$. Then the equation (4) relates to the following diffusion equation:

$$u_t=\frac{\lambda_1}{\sigma^2}(v-u)-\lambda_2\left(1-\frac{v}{u}\right)+\mu\frac{u_{xx}u_y^2-2u_xu_yu_{xy}+u_x^2u_{yy}}{\left(u_x^2+u_y^2\right)^{\frac{3}{2}}}=0, \qquad (5)$$

3. MODEL DISCRETIZATION

The discretized form of the equation (5) can be written as following:

$$u_{ij}^{k+1} = u_{ij}^{k} + \xi\left(\frac{\lambda_1}{\sigma^2}\left(v_{ij} - u_{ij}^{k}\right) - \lambda_2\left(1 - \frac{v_{ij}}{u_{ij}^{k}}\right) + \mu\varphi_{ij}^{k}\right), \qquad (6)$$

where

$$\varphi_{ij}^{k} = \frac{\nabla_{xx}\left(u_{ij}^{k}\right)\left(\nabla y\left(u_{ij}^{k}\right)\right)^2 - 2\nabla_x\left(u_{ij}^{k}\right)\nabla_y\left(u_{ij}^{k}\right)\nabla_{xy}\left(u_{ij}^{k}\right) + \left(\nabla_x\left(u_{ij}^{k}\right)\right)^2\nabla_{yy}\left(u_{ij}^{k}\right)}{\left(\left(\nabla_x\left(u_{ij}^{k}\right)\right)^2 + \left(\nabla_y\left(u_{ij}^{k}\right)\right)^2\right)^{3/2}},$$

$$u_{0j}^{k} = u_{1j}^{k}; u_{N_1+1,j}^{k} = u_{N_1,j}^{k}; u_{i0}^{k} = u_{i1}^{k}; u_{i,N_2+1}^{k} = u_{i,N_2}^{k}; i = 1, \dots, N_1; j = 1, \dots, N_2;$$
$$k = 0,1, \dots, K; \Delta x = \Delta y = 1; 0 < \xi < 1.$$

4. FINDING OPTIMAL PARAMETERS

$$u_{ij}^{k+1} = u_{ij}^{k} + \xi\left(\frac{\lambda_1}{\sigma^2}\left(v_{ij} - u_{ij}^{k}\right) - \lambda_2\left(1 - \frac{v_{ij}}{u_{ij}^{k}}\right) + \mu\varphi_{ij}^{k}\right), \qquad (6)$$

We can use the procedure (6) to perform image denoising. In this procedure, values of parameters $\lambda_1, \lambda_2, \mu, \sigma$ need to be given. In some cases, we have to define these parameters to perform image denoising automatically. Then parameters $\lambda_1, \lambda_2, \mu$ in process (6) need to be changed into $\lambda_1^k, \lambda_2^k, \mu^k$ for each step k. So we obtain new procedure that alows us to calculate values of these parameters automatically in iteration steps.

4. FINDING OPTIMAL PARAMETERS

Optimal parameters λ_1 and λ_2

Let (u, τ) be a solution of the problem (2). Then we get the conditon:

$$\frac{\partial L(u,\tau)}{\partial u} = 0.$$

This condition gives us the optimal parameters λ_1 and λ_2:

$$\lambda_1 = \frac{\int_\Omega \left(1 - \frac{v}{u}\right) dxdy}{\frac{1}{\sigma^2}\int_\Omega (v - u) dxdy + \int_\Omega \left(1 - \frac{v}{u}\right) dxdy}, \lambda_2 = 1 - \lambda_1.$$

4. FINDING OPTIMAL PARAMETERS

Optimal parameters λ_1 and λ_2

And their discretized forms:

$$\lambda_1^k = \frac{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(1 - \frac{v_{ij}}{u_{ij}^k}\right)}{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(\frac{v_{ij} - u_{ij}^k}{\sigma^2} + 1 - \frac{v_{ij}}{u_{ij}^k}\right)}, \lambda_2^k = 1 - \lambda_1^k,$$

where $k = 0,1, \ldots, K$.

4. FINDING OPTIMAL PARAMETERS

Optimal parameter $\boldsymbol{\mu}$

In order to find an optimal parameter μ, we multiply (3) by $(u - v)$ and integrate by parts over Ω. Finally, we obtain the formula to find the optimal parameter μ:

$$\mu = \frac{\int_\Omega \left(-\frac{\lambda_1}{\sigma^2}(u-v)^2 - \lambda_2 \frac{(u-v)^2}{u} \right) dxdy}{\int_\Omega \left(\sqrt{u_x^2 + u_y^2} - \frac{u_x v_x + u_y v_y}{\sqrt{u_x^2 + u_y^2}} \right) dxdy}.$$

4. FINDING OPTIMAL PARAMETERS

Optimal parameter $\boldsymbol{\mu}$

Its discretized form:

$$\mu^k = \frac{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\left(-\frac{\lambda_1^k}{\sigma^2}\left(u_{ij}^k - v_{ij}\right)^2 - \lambda_2\frac{\left(u_{ij}^k - v_{ij}\right)^2}{u_{ij}^k}\right)}{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\eta_{ij}^k},$$

where

a) $$\eta_{ij}^k = \sqrt{\left(\nabla_x\left(u_{ij}^k\right)\right)^2 + \left(\nabla_y\left(u_{ij}^k\right)\right)^2} - \frac{\nabla_x\left(u_{ij}^k\right)\nabla_x\left(v_{ij}\right) + \nabla_y\left(u_{ij}^k\right)\nabla_y\left(v_{ij}\right)}{\sqrt{\left(\nabla_x\left(u_{ij}^k\right)\right)^2 + \left(\nabla_y\left(u_{ij}^k\right)\right)^2}}.$$

4. FINDING OPTIMAL PARAMETERS

Optimal parameter $\boldsymbol{\mu}$

Its discretized form:

$$\mu^k = \frac{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\left(-\frac{\lambda_1^k}{\sigma^2}\left(u_{ij}^k - v_{ij}\right)^2 - \lambda_2^k \frac{\left(u_{ij}^k - v_{ij}\right)^2}{u_{ij}^k}\right)}{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\eta_{ij}^k},$$

where

b) $$\nabla_x\left(u_{ij}^k\right) = \frac{u_{i+1,j}^k - u_{i-1,j}^k}{2\Delta x}, \nabla_y\left(u_{ij}^k\right) = \frac{u_{i,j+1}^k - u_{i,j-1}^k}{2\Delta y},$$

$$u_{0j}^k = u_{1j}^k, u_{N_1+1,j}^k = u_{N_1}^k, u_{i0}^k = u_{i1}^k, u_{i,N_2+1}^k = u_{i,N_2}^k$$

4. FINDING OPTIMAL PARAMETERS

Optimal parameter $\boldsymbol{\mu}$

Its discretized form:

$$\mu^k = \frac{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\left(-\frac{\lambda_1^k}{\sigma^2}\left(u_{ij}^k - v_{ij}\right)^2 - \lambda_2^k \frac{\left(u_{ij}^k - v_{ij}\right)^2}{u_{ij}^k}\right)}{\sum_{i=1}^{N_1}\sum_{j=1}^{N_2}\eta_{ij}^k},$$

where

c)

$$\nabla_x(v_{ij}) = \frac{v_{i+1,j} - v_{i-1,j}}{2\Delta x}, \nabla_y(v_{ij}) = \frac{v_{i,j+1} - v_{i,j-1}}{2\Delta y},$$

$$v_{0j} = v_{1j}, v_{N_1+1,j} = v_{N_1,j}, v_{i0} = v_{i1}, v_{i,N_2+1} = v_{i,N_2},$$

$$i = 1, \ldots, N_1; j = 1, \ldots, N_2; k = 0,1, \ldots, K; \Delta x = \Delta y = 1.$$

4. FINDING OPTIMAL PARAMETERS

Optimal parameter σ

In order to evaluate this parameter σ, we use the result of Immerker:

$$\sigma = \frac{\sqrt{\frac{\pi}{2}}}{6(N_1 - 2)(N_2 - 2)} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} |u_{ij} * \Lambda| \, ,$$

where

$$\Lambda = \begin{pmatrix} 1 & -2 & 1 \\ -2 & 4 & -2 \\ 1 & -2 & 1 \end{pmatrix}$$ - is the mask of an image,

Operator $*$ - convolution operator.

We just evaluate this parameter at first time of the iteration process.

5. IMAGE QUALITY EVALUATION

$$Q_{MSE} = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(u_{ij} - v_{ij}\right)^2, \qquad Q_{PSNR} = 10 \lg\left(\frac{N_1 N_2 L^2}{\sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(u_{ij} - v_{ij}\right)^2} \right),$$

$$Q_{SSIM} = \frac{(2\overline{u}\,\overline{v} + C_1)(2\sigma_{uv} + C_2)}{(\overline{u}^2 + \overline{v}^2 + C_1)(\sigma_u^2 + \sigma_v^2 + C_2)},$$

where

$$\overline{u} = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} u_{ij}\,, \overline{v} = \frac{1}{N_1 N_2} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} v_{ij}\,, \sigma_u^2 = \frac{1}{N_1 N_2 - 1} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(u_{ij} - \overline{u}\right)^2,$$

$$\sigma_v^2 = \frac{1}{N_1 N_2 - 1} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} \left(v_{ij} - \overline{v}\right)^2, \sigma_{uv} = \frac{1}{N_1 N_2 - 1} \sum_{i=1}^{N_1} \sum_{j=1}^{N_2} (u_{ij} - \overline{u})(v_{ij} - \overline{v}),$$

$$C_1 = (K_1 L)^2, C_2 = (K_2 L)^2, K_1 \ll 1, K_2 \ll 1,$$

For example, $K_1 = K_2 = 10^{-6}, L$ – image intensity, $L = 2^8 - 1 = 255$.

6. LINEAR COMBINATION vs SUPERPOSITION

In practice, the mixed Poisson-Gaussian noise is usually their superposition. This is caused by natural physical processes of image formation: in many cases Poisson noise is added into image first and Gaussian noise is added later.

In order to remove the mixed noise, we assume the superposition is equivalent to some unknown linear combination of Poisson and Gaussian noises. In series of experiments before, we showed our model "feels" well the wide range of proportion of two types of noises in linear combination including Poisson and Gaussian noises separately. Our model proves to be better than some other models (Wiener and median filters, Beltrami regularization, ROF, modified ROF).

In fact, the linear model appears to be the good basis to remove superposition of noises. Additionally, this model stays relevant to reversed superposition of noises in some other real situations of light radiation [(1)], since it doesn't matter what superposition to evaluate.

------

(1) **Klauder, J.R., Sudarshan, E.C.G. 2006.** ***Fundamentals of Quantum Optics (Dover Books on Physics)*****. Dover Publications.**

7. COMPARISON WITH PURE-LET METHOD

The PURE-LET method is designed to remove the mixed noise (superposition of Poisson and Gaussian noises) based on the strong theoretical basis by Lusier F., Blu T., Unser M. (Image Denoising in Mixed Poisson–Gaussian Noise//IEEE Transactions on image processing, Vol. 20, No. 3, 2011, P. 696-708).

We compare denoising results for two cases: the linear combination of Poisson and Gaussian noises and the superposition of noises. The tests will be applied for artificial images and real images with artificial noises.

7. COMPARISON WITH PURE-LET METHOD

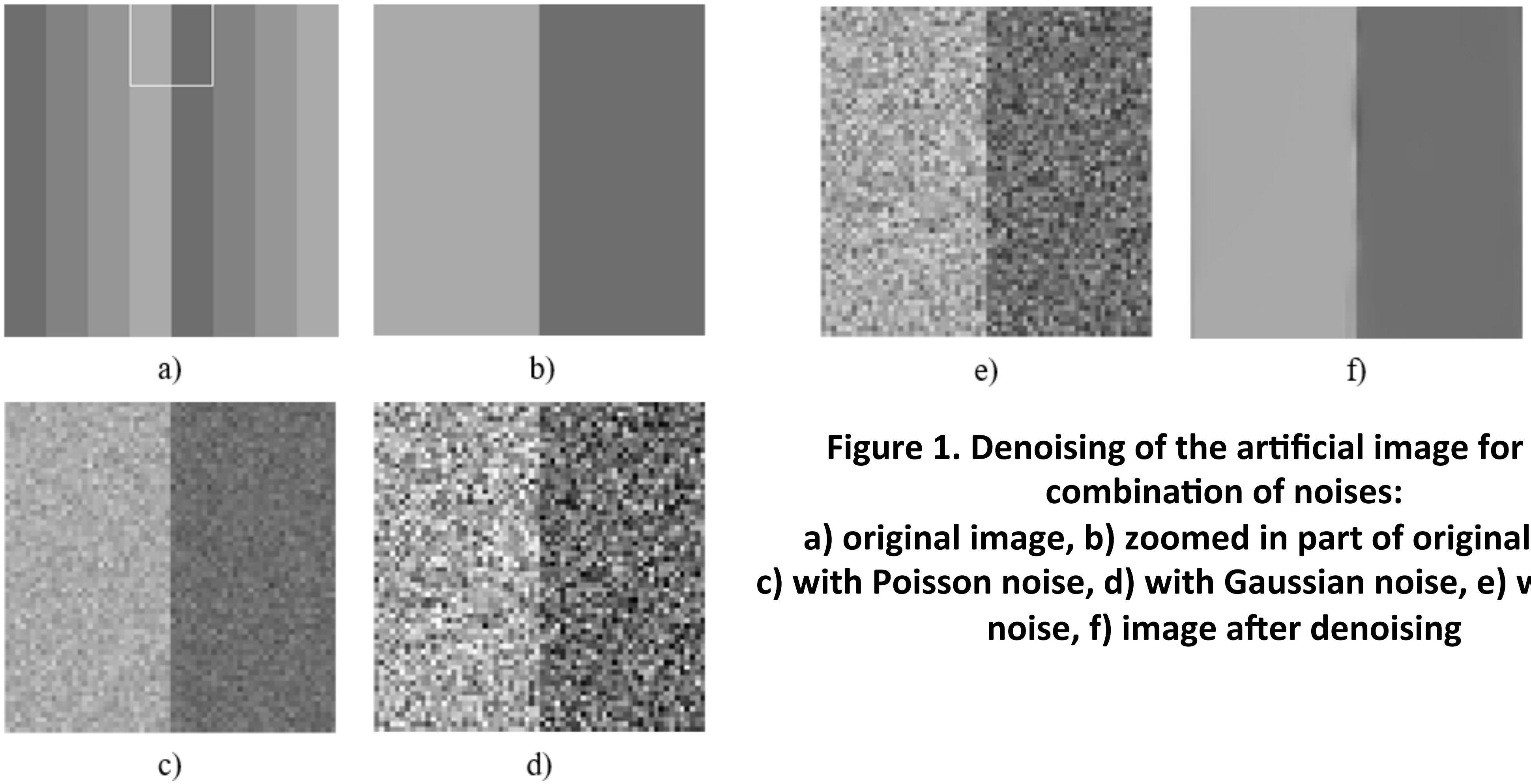


Figure 1. Denoising of the artificial image for linear combination of noises:
a) original image, b) zoomed in part of original image,
c) with Poisson noise, d) with Gaussian noise, e) with mixed noise, f) image after denoising

7. COMPARISON WITH PURE-LET METHOD

1. EXPERIMENTS FOR ARTIFICIAL IMAGE WITH ARTIFICIAL NOISE

- We use artificial image with artificial mixed noise for the first test. The image size is 256x256, it includes eight bars (Figures 1a, 2a). Other images (Figures 1b-1f, 2b-2f) show the zoomed in part of it.
- The artificial noise will generated by two methods: by linear combination of Poisson and Gaussian noises; by superposition of Poisson and Gaussian noises.
- For both cases, we add Poisson noise first. The variance of Poisson noise is calculated as average value $\overline{\sigma_2} = 1/4\left(\sqrt{110} + \sqrt{130} + \sqrt{150} + \sqrt{170}\right)$**=11.7939.** If the value of pixel after adding Poisson noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(2)} = u_{ij}$. For this image, there is no number of pixels, that are out of this interval.

7. COMPARISON WITH PURE-LET METHOD

1. EXPERIMENTS FOR ARTIFICIAL IMAGE WITH ARTIFICIAL NOISE

- For Gaussian noise. We consider that the variance of Gaussian noise is four times greater than the variance of Poisson noise $\sigma_1 = 4\sigma_2 = 47.1757$.
- For linear combination, we denote the intensity function of this Gaussian noisy image as $v^{(1)}$. As we explain above, the values of intensity function $v^{(1)}$ also need to be between 0 to 255. If the value of pixel after adding Gaussian noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(1)} = u_{ij}$. In this case, there are 1075 pixels out of this interval (1.6403%). For final noisy image (Figure 1e), we use linear combination to combine Gaussian noisy image $v^{(1)}$ and Poisson noisy image $v^{(2)}$ with the coefficients of linear combination that are 0.6 for Gaussian noisy image and 0.4 for Poisson noisy image: $v = 0.6v^{(1)} + 0.4v^{(2)}$. Then we obtain the proportion λ_1/λ_2=(0.6*47.1757)/(0.4*11.7939)=6/1. That means λ_1=0.8571, λ_2=0.1429. The values of Q_{PSNR}, Q_{MSE}, and Q_{SSIM} of final noisy image are, respectively, 19.4291, 741.5963, and 0.1073.

7. COMPARISON WITH PURE-LET METHOD

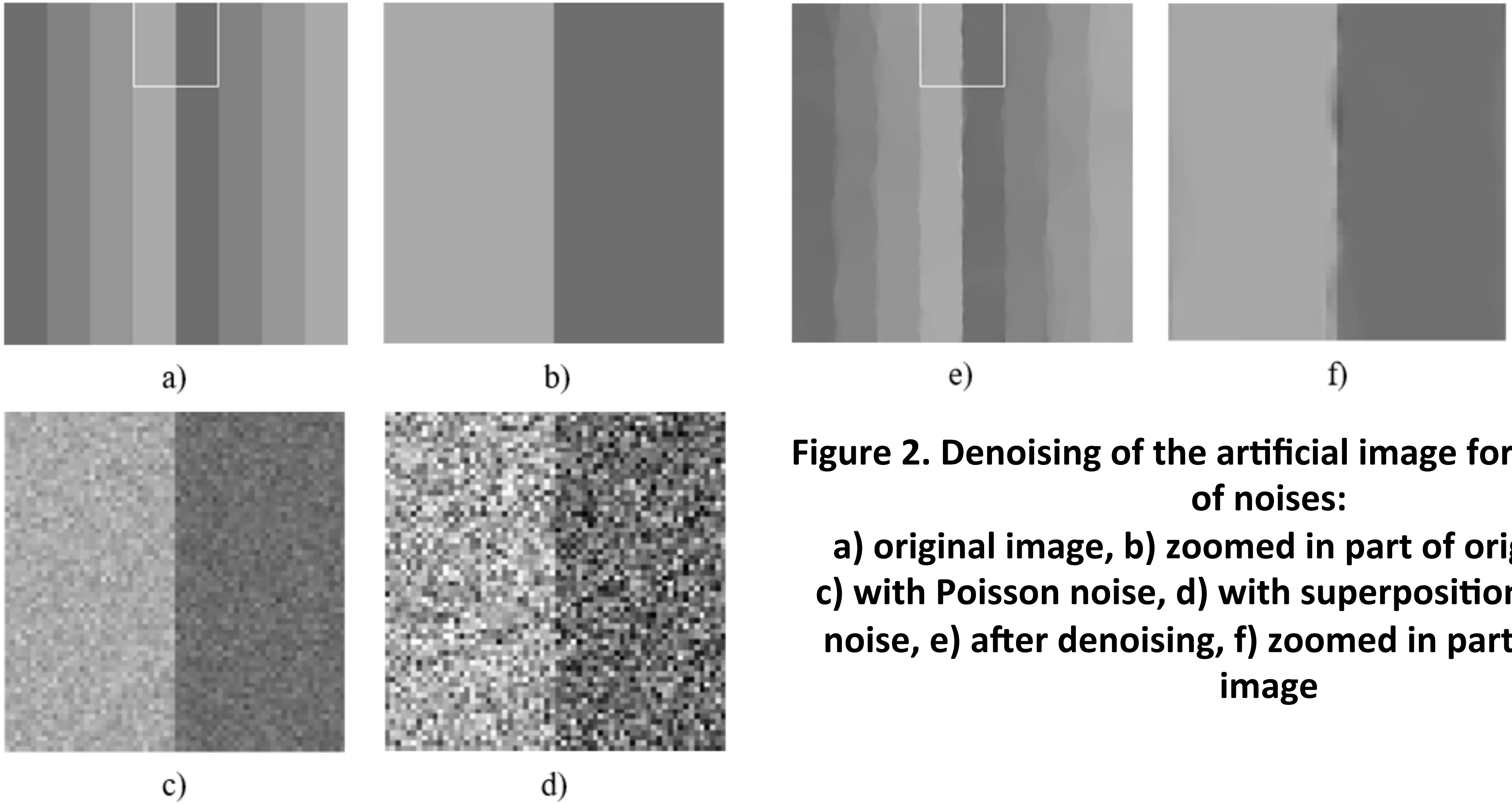


Figure 2. Denoising of the artificial image for superposition of noises:
a) original image, b) zoomed in part of original image, c) with Poisson noise, d) with superposition of Gaussian noise, e) after denoising, f) zoomed in part of denoised image

7. COMPARISON WITH PURE-LET METHOD

1. EXPERIMENTS FOR ARTIFICIAL IMAGE WITH ARTIFICIAL NOISE

- For superposition of noises, we add Gaussian noise into Poisson noisy image. We denote the intensity function of this Gaussian noisy image as $v^{(1)}$. As we explain above, the values of intensity function $v^{(1)}$ also need to be between 0 to 255. If the value of pixel after adding Gaussian noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(1)} = v_{ij}^{(2)}$. In this case, there are 1220 pixels out of this interval (1.8616%). The final noisy image (Figure 2e) is also the Gaussian noisy image $v = v^{(1)}$. The calculation for λ_1 and λ_2 is very difficult, so we use the algorithm with automatically defined parameters to find them. The values of Q_{PSNR}, Q_{MSE}, and Q_{SSIM} of final noisy image are, respectively, 14.9211, 2093.9827, and 0.0439.
- Tables 1 and 2 show the denoising result for linear combination of noises and the superposition of noises for the artificial image with artificial mixed noise.

7. COMPARISON WITH PURE-LET METHOD

Table 1. Quality comparison of noise removal methods for the image with linear combination of noises.

Artificial Image	Q_{PSNR}	Q_{SSIM}	Q_{MSE}
Noisy	19.4291	0.1073	741.5963
ROF	***34.1236***	***0.8978***	25.1606
Modified ROF	32.4315	0.8703	37.8791
PURE-LET for mixed noise	***33.0309***	***0.9277***	32.3587
Proposed method with automatically defined parameters λ_1=0.8414, λ_2=0.1586, μ = 0.5112, σ = 41.0314	**41.0998**	**0.9840**	**5.0478**

Table 2. Quality comparison of noise removal methods for the image with superposition of noises.

Artificial Image	Q_{PSNR}	Q_{SSIM}	Q_{MSE}
Noisy	14.9211	0.0439	2093.983
ROF	31.2913	0.8346	48.3008
Modified ROF	30.5471	0.8232	56.5601
PURE-LET for mixed noise	33.9889	0.9298	25.9534
Proposed method with automatically defined parameters λ_1=0.8014, λ_2=0.1986, μ = 0.4812, σ = 40.0314	**37.3366**	**0.9677**	**12.0066**

7. COMPARISON WITH PURE-LET METHOD

2. EXPERIMENTS FOR REAL IMAGE WITH ARTIFICIAL NOISE

- We use real image with artificial mixed noise for the second test. The image is human's skull image with size is 256x256 (Figures 3a, 4a). Other images (Figures 3b-3f, 4b,-4f) show the zoomed out part of it.
- The artificial noise will generated by two methods: by linear combination of Poisson and Gaussian noises; by superposition of Poisson and Gaussian noises.
- For both cases, we add Poisson noise first. The variance of Poisson noise is calculated as average value $\overline{\sigma_2}$**=9.0882.** If the value of pixel after adding Poisson noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(2)} = u_{ij}$. For this image, there is no number of pixels, that are out of this interval.

7. COMPARISON WITH PURE-LET METHOD

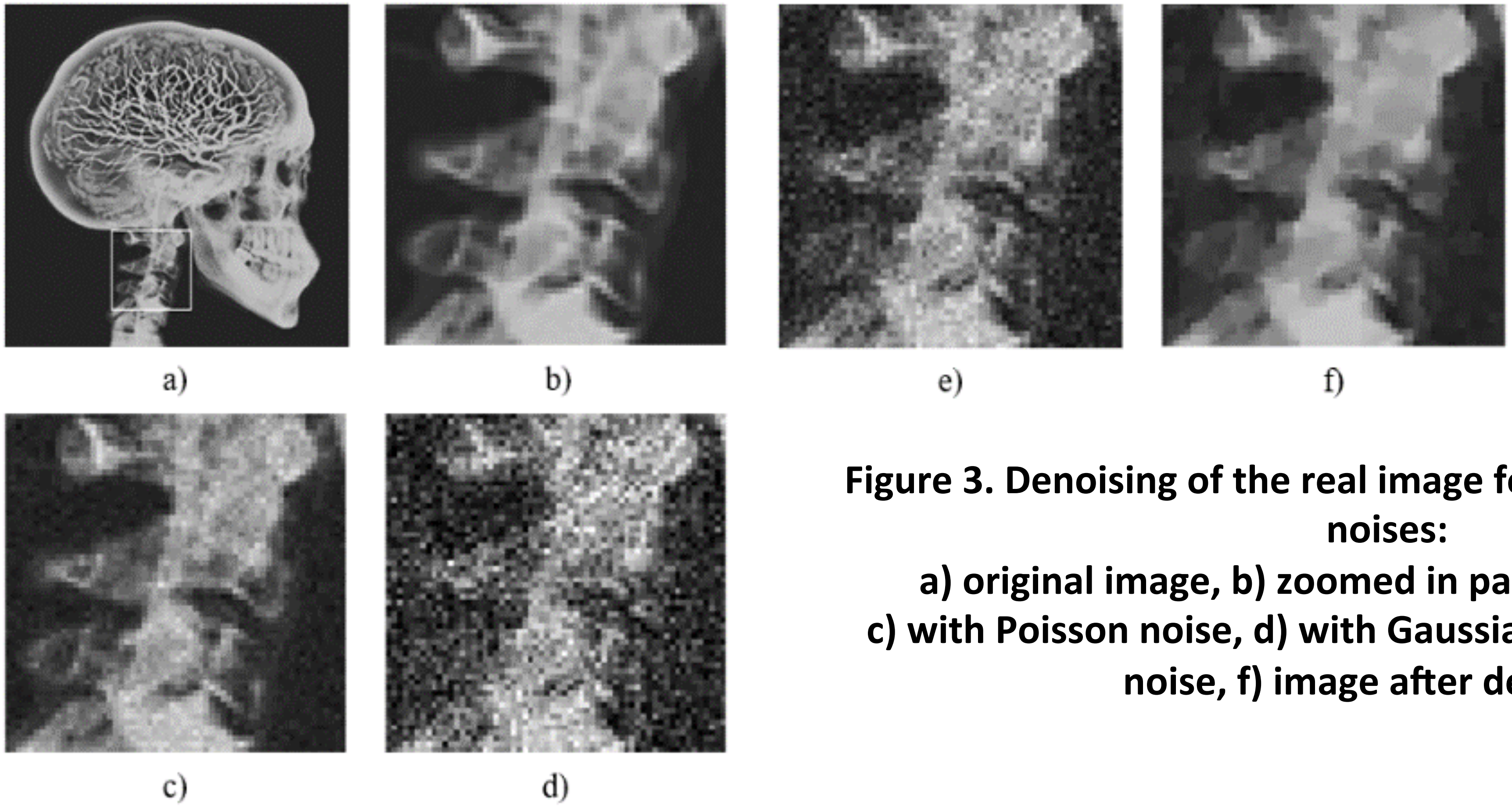


Figure 3. Denoising of the real image for linear combination of noises:
a) original image, b) zoomed in part of original image,
c) with Poisson noise, d) with Gaussian noise, e) with mixed noise, f) image after denoising

7. COMPARISON WITH PURE-LET METHOD

2. EXPERIMENTS FOR REAL IMAGE WITH ARTIFICIAL NOISE

- For Gaussian noise we consider that the variance of Gaussian noise is four times greater than the variance of Poisson noise $\sigma_1 = 4\sigma_2$ =**36.3529.**
- For linear combination, we denote the intensity function of this Gaussian noisy image as $v^{(1)}$**.** As we explain above, the values of intensity function $v^{(1)}$ also need to be between 0 to 255. If the value of pixel after adding
- Gaussian noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(1)} = u_{ij}$**.** In this case, there are 5355 pixels out of this interval (8.1711%). For final noisy image (Figure 3e), we use linear combination to combine Gaussian noisy image $v^{(1)}$ and Poisson noisy image $v^{(2)}$ with the coefficients of linear combination that are 0.6 for Gaussian noisy image and 0.4 for Poisson noisy image: $v = 0.5v^{(1)} + 0.5v^{(2)}$**.** Then we obtain the proportion λ_1/λ_2=(0.5*36.3529)/(0.5*9.0882)=4/1. That means λ_1=0.8, λ_2=0.2. The values of Q_{PSNR}, Q_{MSE}, and Q_{SSIM} of final noisy image are, respectively, 23.6878, 278.1619, and 0.5390.

7. COMPARISON WITH PURE-LET METHOD

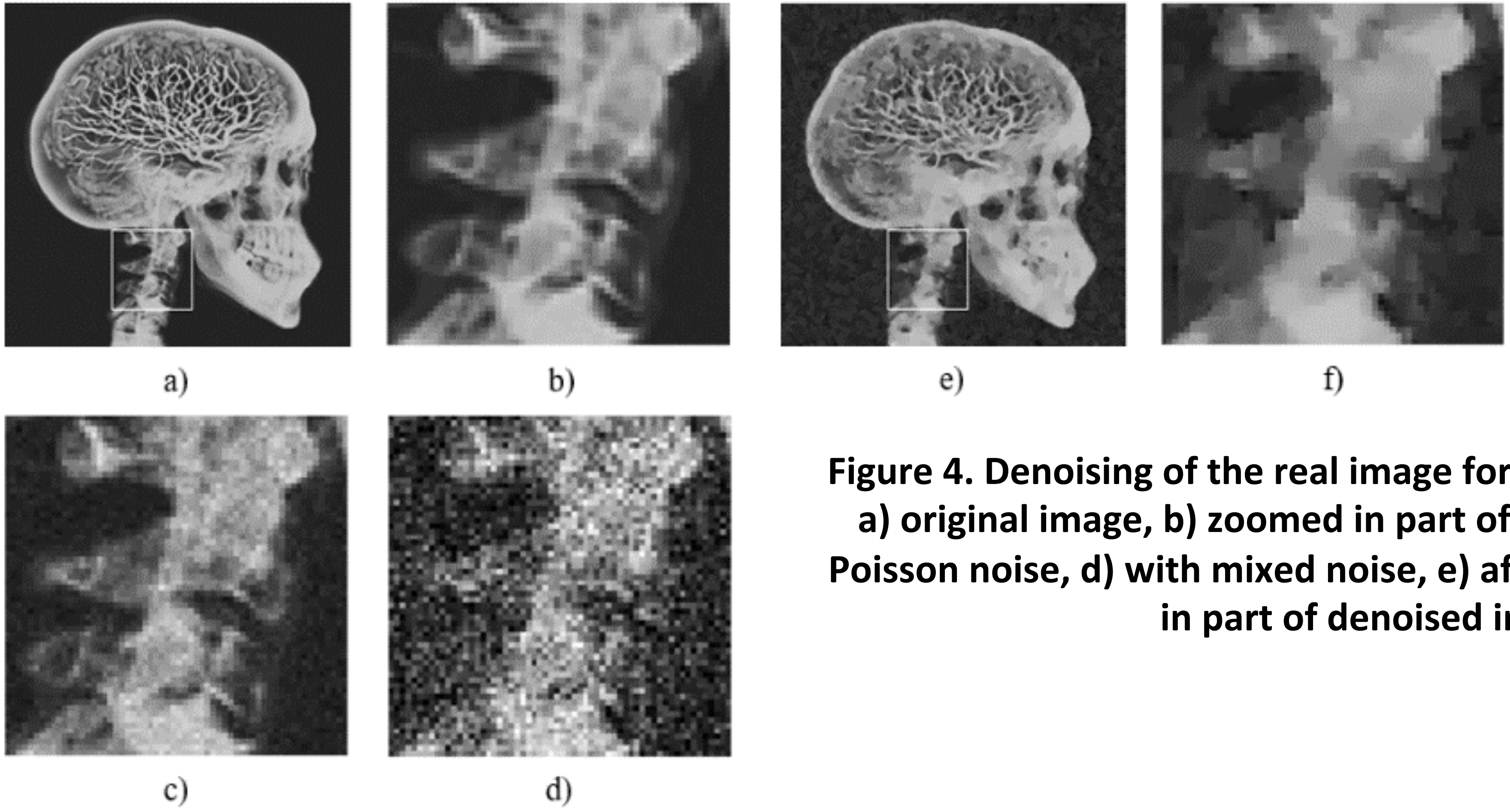


Figure 4. Denoising of the real image for superposition of noises: a) original image, b) zoomed in part of original image, c) with Poisson noise, d) with mixed noise, e) after denoising, f) zoomed in part of denoised image

7. COMPARISON WITH PURE-LET METHOD

2. EXPERIMENTS FOR REAL IMAGE WITH ARTIFICIAL NOISE

- For superposition of noises, we add Gaussian noise into Poisson noisy image. We denote the intensity function of this Gaussian noisy image as $v^{(1)}$. As we explain above, the values of intensity function $v^{(1)}$. also need to be between 0 to 255. If the value of pixel after adding Gaussian noise is out of the interval from 0 to 255, it needs to be reset to $v_{ij}^{(1)} = v_{ij}^{(2)}$. In this case, there are 5621 pixels out of this interval (8.5770%). The final noisy image (Figure 4e) is also the Gaussian noisy image $v = v^{(1)}$. The calculation for λ_1 and λ_2 is very difficult, so we use the algorithm with automatically defined parameters to find them. The values of Q_{PSNR}, Q_{MSE}, and Q_{SSIM} of final noisy image are, respectively, 17.8071, 1077.3831, and 0.3242.
- Tables 3 and 4 show the denoising result for linear combination of noises and the superposition of noises for the real image with artificial mixed noise.

7. COMPARISON WITH PURE-LET METHOD

Table 3. Quality comparison of noise removal methods for the image with linear combination of noises.

Real Image	Q_{PSNR}	Q_{SSIM}	Q_{MSE}
Noisy	23.6878	0.5390	278.1619
ROF	27.3974	0.8295	118.3975
Modified ROF	25.5644	0.7513	197.5403
PURE-LET for mixed noise	25.7781	0.8105	191.0341
Proposed method with automatically defined parameters λ_1=0.7804, λ_2=0.2196, μ = 0.0512, σ = 34.2311	**27.6039**	**0.8325**	**112.8984**

Table 4. Quality comparison of noise removal methods for the image with superposition of noises.

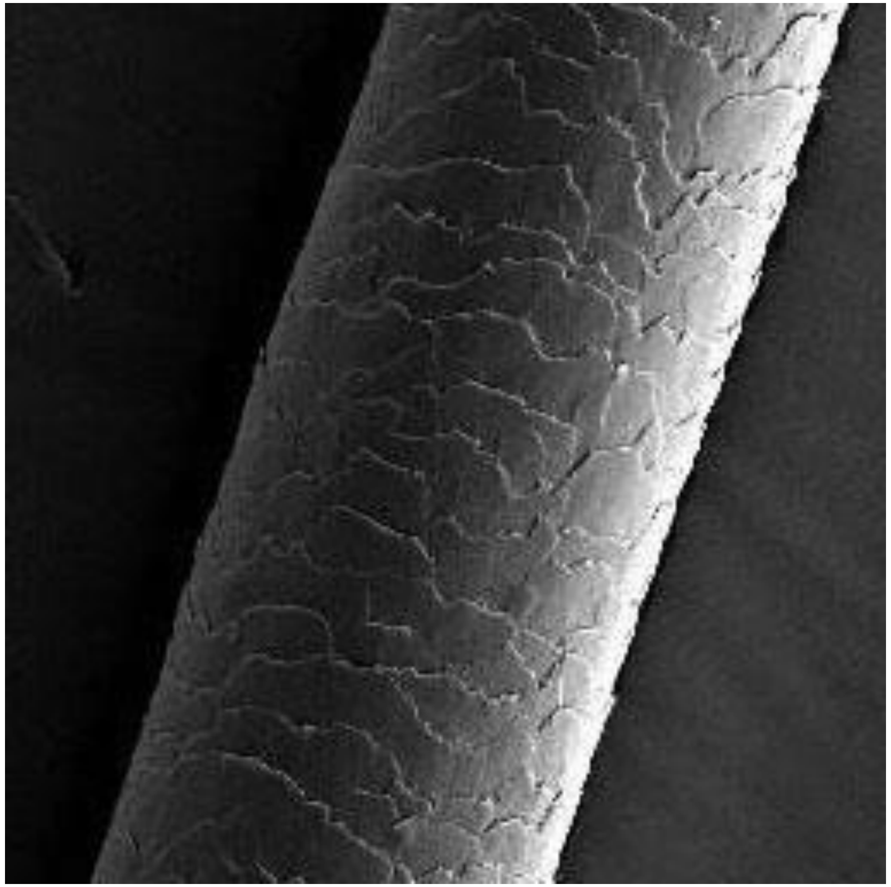
Real Image	Q_{PSNR}	Q_{SSIM}	Q_{MSE}
Noisy	17.8077	0.3242	1077.3831
ROF	23.1936	0.7062	311.6856
Modified ROF	23.0413	0.7033	319.3831
PURE-LET for mixed noise	23.6278	0.7072	282.0349
Proposed method with automatically defined parameters λ_1=0.7704, λ_2=0.2296, μ = 0.1102, σ = 36.3412	**23.7292**	**0.7094**	**275.5229**

REAL NOISE DATA (ALBERTA UNIVERSITY, CANADA)

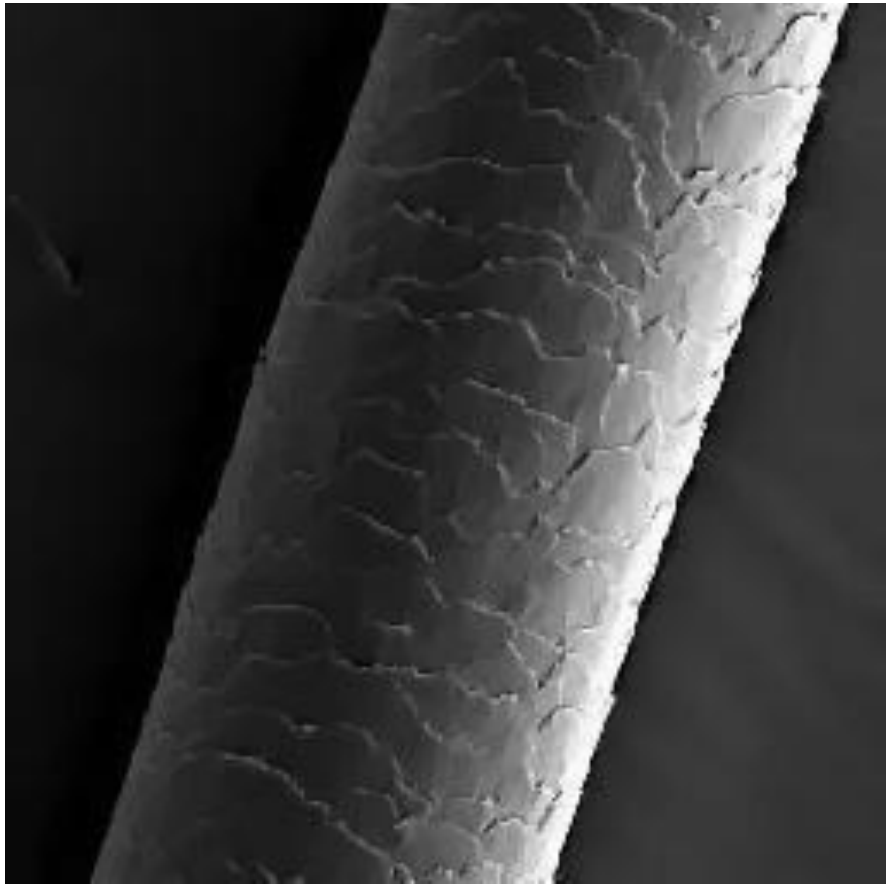
Original No Reference Image

Denoised Image (*BRISQUE*)

REAL NOISE DATA (ALBERTA UNIVERSITY, CANADA)

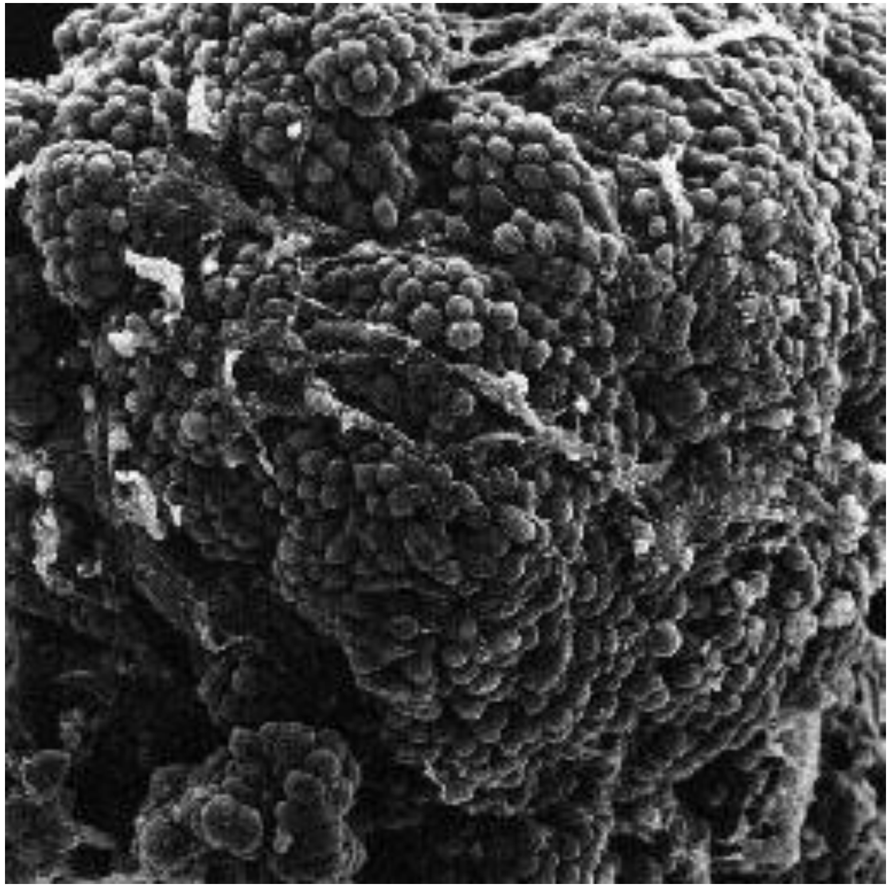


Original No Reference Image

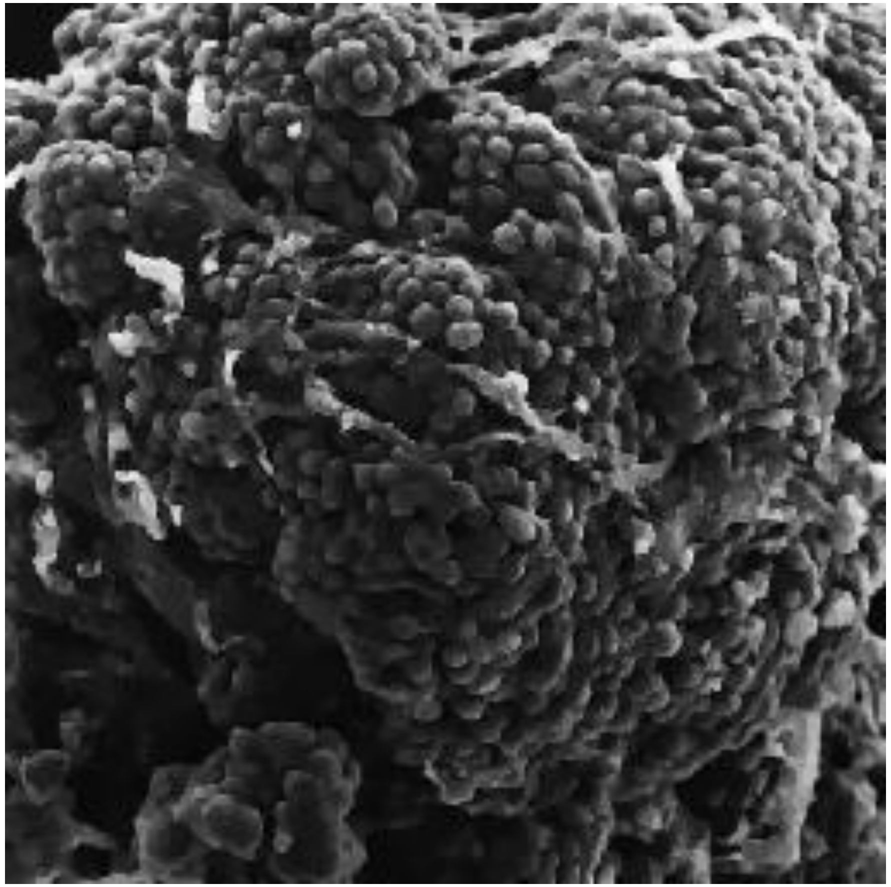


Denoised Image (*BRISQUE*)

REAL NOISE DATA (ALBERTA UNIVERSITY, CANADA)



Original No Reference Image



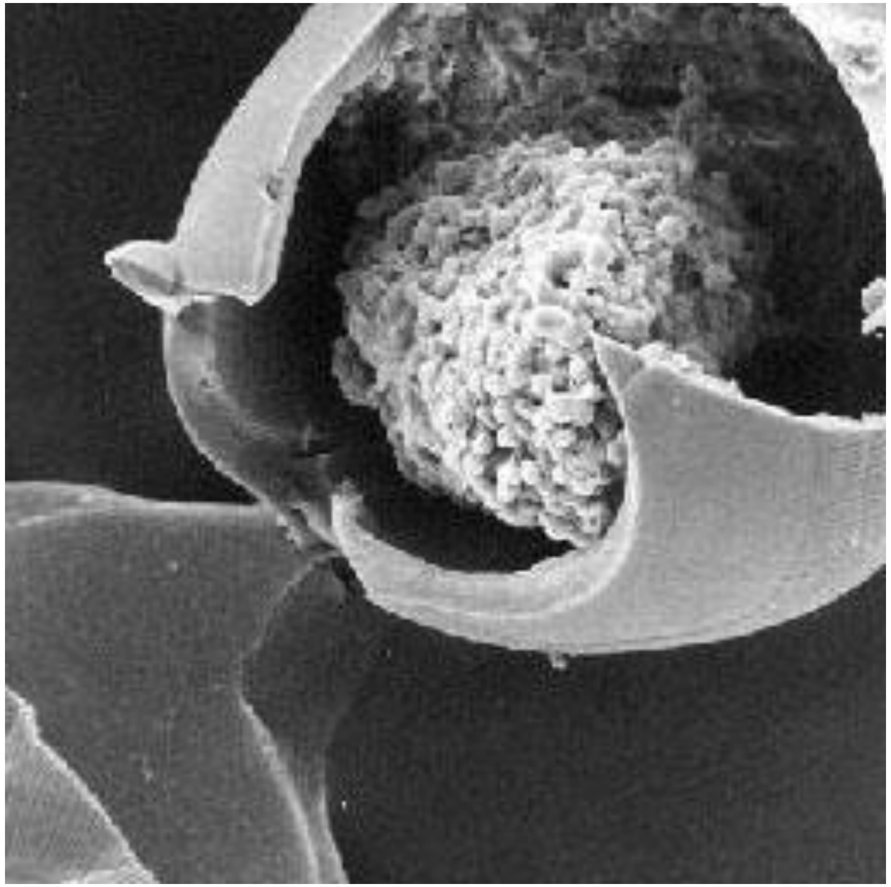
Denoised Image (*BRISQUE*)

REAL NOISE DATA (ALBERTA UNIVERSITY, CANADA)

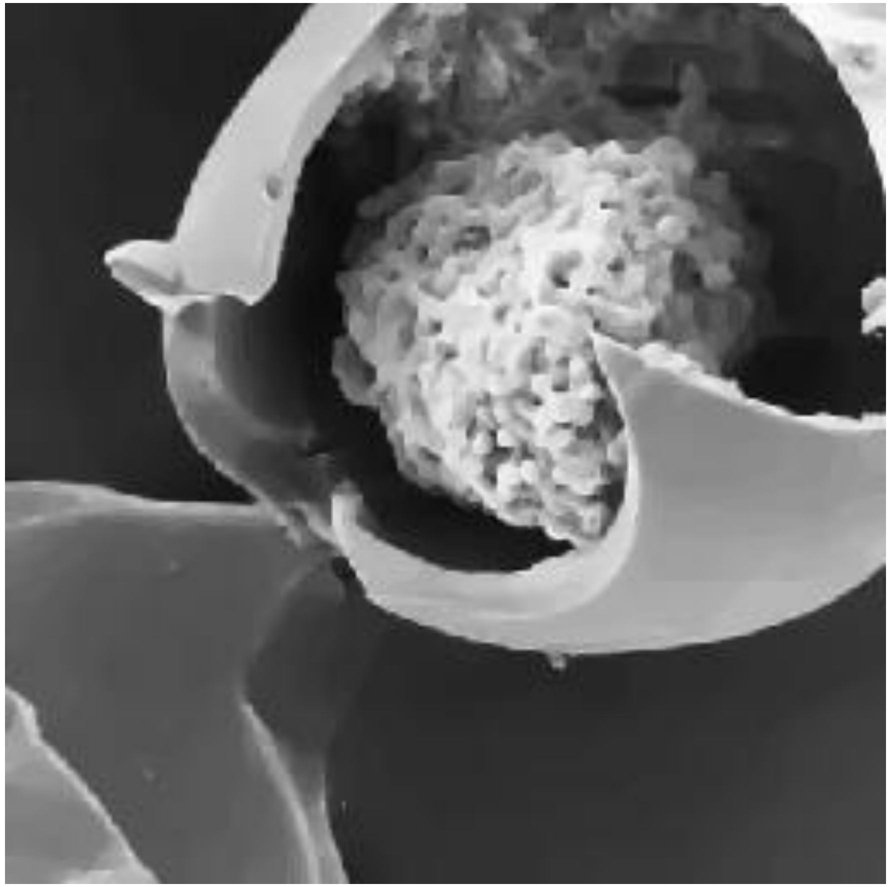
Original No Reference Image

Denoised Image (*BRISQUE*)

REAL NOISE DATA (ALBERTA UNIVERSITY, CANADA)

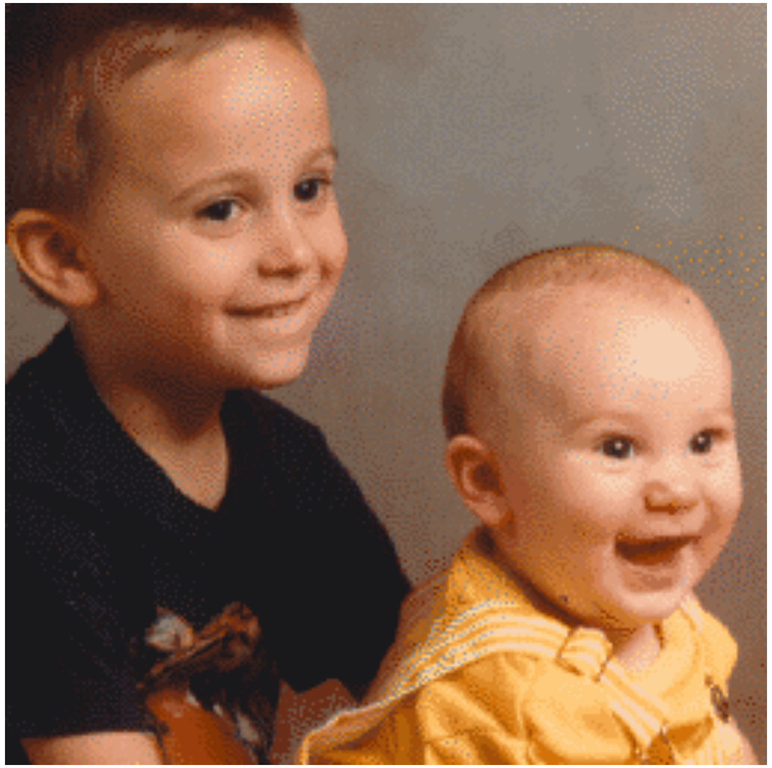


Original No Reference Image



Denoised Image (*BRISQUE*)

RGB-IMAGE (MATLAB DB)



Original

Noised by P-G superposition

Denoised

RGB-IMAGE (MATLAB DB)

Original

Noised by P-G superposition

Denoised

CONCLUSION

In this report, we proposed a novel method that can effectively remove the mixed Poisson-Gaussian noise. Furthermore, our proposed method can be also used to remove Gaussian or Poisson noises separately. This method is based on variational approach.

The denoising result depends on values of coefficients of linear combination λ_1 and λ_2. These values can be set manually or can be defined automatically. When processing real images, we can use the proposed method with automatically defined parameters. The proposed model “feels” well the wide range of proportion of two noises.

In fact, the linear model appears to be the good basis to remove superposition of noises. Additionally, this model stays relevant to reversed superposition of noises in some other real situations of light radiation, since it doesn’t matter what superposition to evaluate. We consider the superposition of noises is equivalent to some unknown linear combination.

Our simple low-parametric model gives better results than PURE-LET in many real cases, because the strong theoretically based PURE-LET is the multiple-parametric model, depending on the quality of its parameters evaluating. Additionally, PURE-LET isn’t relevant to reversed case of superposition.

THANK YOU FOR ATTENTION!