EM+TV and SART+TV for image reconstruction

Ming Yan
University of California, Los Angeles

May 17, 2010

1 Radon Transform

In order to use algebraic reconstruction methods, we have the following system of linear equations,

$$Ax = b, \tag{1}$$

here, A is the matrix defining the radon transform, x is the image, and b is the measurements. The problem is to reconstruct x from b. The matrix A can be derived by Siddon's algorithm. For constructing A, there are many articles on this, we will skip this for this time. For 3D image reconstruction, the matrix A will be very large, and it is unable to store it. We have to define two functions one is Ax, another is $A^T y$.

2 EM+TV

For EM(Expectation Maximization)+TV(Total Variation), we assume that the noise in the measurements are poisson noise and independent. Then

$$P(b|Ax) = \prod_{i=1}^{M} e^{-a_i x} \frac{(a_i x)^{b_i}}{b_i!}, \tag{2}$$

where a_i is the i^{th} row of A.

The optimization problem is

$$\begin{aligned} &\underset{x}{\text{minimize}} \int |\nabla x| - \alpha \log P \\ &\text{subject to } x \geq 0, \end{aligned} \tag{3}$$

here ∇x is a vector, and each component is a linear operation on x. When we use ∇, we consider x to be 2D or 3D image. Otherwise, x is considered to be a vector, except stated.

Plugging into the probability gives us

$$\begin{aligned} &\underset{x}{\text{minimize}} \int |\nabla x| + \alpha \sum_{i=1}^{M} (a_i x - b_i \log(a_i x)) \\ &\text{subject to } x \geq 0. \end{aligned} \tag{4}$$

The KKT condition of this is

$$-\text{div}(\frac{\nabla x}{|\nabla x|})_j + \alpha \sum_{i=1}^{M} (a_{ji}(1 - \frac{b_i}{a_i x})) - y_j = 0 \qquad j = 1, \cdots, N \tag{5}$$

$$y \geq 0 \tag{6}$$

$$x \geq 0 \tag{7}$$

$$y^T x = 0. \tag{8}$$

It becomes

$$-\frac{x_j}{a_j^T \overrightarrow{1}} \text{div}(\frac{\nabla x}{|\nabla x|})_j + \alpha \frac{\sum_{i=1}^{M}(a_{ij}(1 - \frac{b_i}{a_i x}))}{a_j^T \overrightarrow{1}} x_j = 0 \qquad j = 1, \cdots, N. \tag{9}$$

or

$$-\frac{x_j}{a_j^T \overrightarrow{1}} \text{div}(\frac{\nabla x}{|\nabla x|})_j + \alpha x_j - \alpha \frac{\sum_{i=1}^{M}(a_{ij}(\frac{b_i}{a_i x}))}{a_j^T \overrightarrow{1}} x_j = 0 \qquad j = 1, \cdots, N. \tag{10}$$

Here a_j is the j^{th} column.

Denote

$$x_j^{EM} = \frac{\sum_{i=1}^{M}(a_{ij}(\frac{b_i}{a_i x}))}{a_j^T \overrightarrow{1}} x_j \tag{11}$$

and this is the EM update.

The KKT condition becomes

$$-\frac{x_j}{a_j^T \overrightarrow{1}} \text{div}(\frac{\nabla x}{|\nabla x|})_j + \alpha x_j - \alpha x_j^{EM} = 0 \qquad j = 1, \cdots, N, \tag{12}$$

and this is the optimality for a TV minimization problem. We can solve it by iterations. For this step, we consider this to be 2D or 3D image. For 2D, we have the new notations $u_{i,j}$.

This is nonlinear equation, so we consider a linearized version

$$- \frac{u_{i,j}^n}{V_{i,j}} \frac{u_{i+1,j}^n - u_{i,j}^{n+1}}{\sqrt{\epsilon + (u_{i+1,j}^n - u_{i,j}^n)^2 + (u_{i,j+1}^n - u_{i,j}^n)^2}} \tag{13}$$

$$+ \frac{u_{i,j}^n}{V_{i,j}} \frac{u_{i,j}^{n+1} - u_{i-1,j}^n}{\sqrt{\epsilon + (u_{i,j}^n - u_{i-1,j}^n)^2 + (u_{i-1,j+1}^n - u_{i-1,j}^n)^2}} \tag{14}$$

$$- \frac{u_{i,j}^n}{V_{i,j}} \frac{u_{i,j+1}^n - u_{i,j}^{n+1}}{\sqrt{\epsilon + (u_{i+1,j}^n - u_{i,j}^n)^2 + (u_{i,j+1}^n - u_{i,j}^n)^2}} \tag{15}$$

$$+ \frac{u_{i,j}^n}{V_{i,j}} \frac{u_{i,j}^{n+1} - u_{i,j-1}^n}{\sqrt{\epsilon + (u_{i+1,j-1}^n - u_{i,j-1}^n)^2 + (u_{i,j}^n - u_{i,j-1}^n)^2}} + \alpha u_{i,j}^{n+1} - \alpha u_{i,j}^{EM} = 0 \tag{16}$$

The code for EM update is as follows:

```
function x = EMCT(A, b, x0, max_iter)

[m, n] = size(A);

x = x0;
stopc = 0;
iter = 0;
sumA = sum(A)';
while ~stopc
    Ax = A*x;
    bAx = b./Ax;
    AtAx = A'*bAx;
    x = x.*AtAx./sumA;
    iter = iter + 1;

    if iter == max_iter || max(abs(A*x-b)) < 1e-8
        stopc = 1;
        fprintf('total iteration %d:\n', iter);
    end
end
```

The code for TV update

```
function u_new = WDenoiseP(f, V, alpha, max_iter)

[M, N] = size(f);
epi2 = 1e-8;
u = f;

for j =1:1:max_iter
 for i=2:1:M-1
    for j=2:1:N-1
        c1(i,j)=u(i,j)/sqrt(epi2+(u(i+1,j)-u(i,j))^2+(u(i,j+1)-u(i,j))^2)/V(i,j);
    end
end
for i=2:1:M-1
    for j=2:1:N-1
        c2(i,j)=u(i,j)/sqrt(epi2+(u(i,j)-u(i-1,j))^2+(u(i-1,j+1)-u(i-1,j))^2)/V(i,j);
    end
end
for i=2:1:M-1
    for j=2:1:N-1
        c3(i,j)=u(i,j)/sqrt(epi2+(u(i+1,j)-u(i,j))^2+(u(i,j+1)-u(i,j))^2)/V(i,j);
    end
end
for i=2:1:M-1
    for j=2:1:N-1
        c4(i,j)=u(i,j)/sqrt(epi2+(u(i+1,j-1)-u(i,j-1))^2+(u(i,j)-u(i,j-1))^2)/V(i,j);
    end
end
```

```
for i=2:1:M-1
    for j=2:1:N-1
        if u(i,j) < epi2
            u1(i,j) = u(i,j);
        else
        u1(i,j)=1/(alpha+c1(i,j)+c2(i,j)+c3(i,j)+c4(i,j))
        *(alpha*f(i,j)+c1(i,j)*u(i+1,j)+c2(i,j)*u(i-1,j)+c3(i,j)*u(i,j+1)+c4(i,j)*u(i,j-1)
        end
    end
end

%%%actualy for the boundary, we can choose vanishing bounday condition.

for i=2:1:M-1
    u1(i,1)=u1(i,2);
    u1(i,N)=u1(i,N-1);
end
for j=1:1:N
    u1(1,j)=u1(2,j);
    u1(M,j)=u1(M-1,j);
end

u = u1;

end
u_new = u;
```

The final code for EM+TV

```
load('raysa.mat','rays') load('sinoa.mat','sino');

index = find(sum(rays,2)>0);
A = rays(index,:);
%W = sum(A);

%W =reshape(W,256,256);

g = sino(index);

V = reshape(sum(A),256,256);
V = full(V);
mu = 5;

fnew = ones(256*256,1);


for i=1:1000
```

```
fnew = EMCT(A, g, fnew, 5);

fnew = reshape(fnew,256,256);
fnew(1,:) = zeros(1,256);

fnew(256,:) = zeros(1,256);
fnew(:,1) = zeros(256,1);

fnew(:,256) = zeros(256,1);


fnew = WDenoiseP(fnew, V, mu, 20);

fnew = reshape(fnew, 256*256,1);

fnew = max(fnew, 0);
fnew = min(fnew, 1);

if mod(i,250)==0
    figure
   imshow(reshape(fnew,256,256),[]);
   figure
   imshow(reshape(fnew,256,256)-phantom(256),[])

end
end
```

3 SART(Simultaneous Algebraic Reconstruction Technique)+TV

Instead of poisson noise, if we assume that the noise is a weighted gaussian noise. Longer the line travels in the object, higher the standard deviation of the measurements. Define

$$w_i = A_{i,+} = \sum_{j=1}^{M} a_{i,j}, \tag{17}$$

$$v_j = A_{+,j} = \sum_{i=1}^{N} a_{i,j}. \tag{18}$$

In addition V and W are diagonal matrices with diagonal element v_j and w_i.

The problem is

$$\underset{x}{\text{minimize}} \int |\nabla x| + \alpha \sum_{i=1}^{M} w_i^{-1}(a_i x - b_i)^2.$$

This is an unconstraint problem, and the solution should satisfy

$$-\mathrm{div}(\frac{\nabla x}{|\nabla x|}) + 2\alpha A^T W^{-1}(Ax - b) = 0. \tag{19}$$

If we denote

$$x^{SART} = x - V^{-1} A^T W^{-1}(Ax - b) \tag{20}$$

we have

$$-\mathrm{div}(\frac{\nabla x}{|\nabla x|}) + 2\alpha V(x - x^{SART)} = 0. \tag{21}$$