# 2015年11月15日 星期日

初始化Face Alignment Master 工程。

# 2015年11月15日 星期日

## tree.cpp 197行

for(int j = 0;j < max\_numfeats\_;j++){

//some problems??? Ming

/\*fixed bug, by Ming, 2015.08.20

double project\_x1 = rotation(0,0) \* candidate\_pixel\_locations(j,0) + rotation(0,1) \* candidate\_pixel\_locations(j,1);

double project\_y1 = rotation(1,0) \* candidate\_pixel\_locations(j,0) + rotation(1,1) \* candidate\_pixel\_locations(j,1);

\*/

double project\_x1 = rotation(0,0) \* candidate\_pixel\_locations(j,0) + rotation(1,0) \* candidate\_pixel\_locations(j,1);

double project\_y1 = rotation(0,1) \* candidate\_pixel\_locations(j,0) + rotation(1,1) \* candidate\_pixel\_locations(j,1);

project\_x1 = scale \* project\_x1 \* bounding\_box[ind\_samples[i]].width / 2.0;

project\_y1 = scale \* project\_y1 \* bounding\_box[ind\_samples[i]].height / 2.0;

int real\_x1 = project\_x1 + current\_shapes[ind\_samples[i]](landmarkID\_,0);

int real\_y1 = project\_y1 + current\_shapes[ind\_samples[i]](landmarkID\_,1);

real\_x1 = max(0.0,min((double)real\_x1,images[ind\_samples[i]].cols-1.0));

real\_y1 = max(0.0,min((double)real\_y1,images[ind\_samples[i]].rows-1.0));

/\*fixed bug, by Ming, 2015.08.20

double project\_x2 = rotation(0,0) \* candidate\_pixel\_locations(j,2) + rotation(0,1) \* candidate\_pixel\_locations(j,3);

double project\_y2 = rotation(1,0) \* candidate\_pixel\_locations(j,2) + rotation(1,1) \* candidate\_pixel\_locations(j,3);

\*/

double project\_x2 = rotation(0,0) \* candidate\_pixel\_locations(j,2) + rotation(1,0) \* candidate\_pixel\_locations(j,3);

double project\_y2 = rotation(0,1) \* candidate\_pixel\_locations(j,2) + rotation(1,1) \* candidate\_pixel\_locations(j,3);

project\_x2 = scale \* project\_x2 \* bounding\_box[ind\_samples[i]].width / 2.0;

project\_y2 = scale \* project\_y2 \* bounding\_box[ind\_samples[i]].height / 2.0;

int real\_x2 = project\_x2 + current\_shapes[ind\_samples[i]](landmarkID\_,0);

int real\_y2 = project\_y2 + current\_shapes[ind\_samples[i]](landmarkID\_,1);

real\_x2 = max(0.0,min((double)real\_x2,images[ind\_samples[i]].cols-1.0));

real\_y2 = max(0.0,min((double)real\_y2,images[ind\_samples[i]].rows-1.0));

densities(j,i) = ((int)(images[ind\_samples[i]](real\_y1,real\_x1))-(int)(images[ind\_samples[i]](real\_y2,real\_x2)));

}

}

## 二、LBFRegressor.cpp 69行

for (int iter = 0;iter<rand\_forest.size();iter++){

int currnode = 0;

int bincode = 1;

for(int i = 0;i<rand\_forest[iter].max\_depth\_-1;i++){

double x1 = rand\_forest[iter].nodes\_[currnode].feat[0];

double y1 = rand\_forest[iter].nodes\_[currnode].feat[1];

double x2 = rand\_forest[iter].nodes\_[currnode].feat[2];

double y2 = rand\_forest[iter].nodes\_[currnode].feat[3];

/\*fixed bug, by Ming, 2015.08.20

double project\_x1 = rotation(0,0) \* x1 + rotation(0,1) \* y1;

double project\_y1 = rotation(1,0) \* x1 + rotation(1,1) \* y1;

\*/

double project\_x1 = rotation(0,0) \* x1 + rotation(1,0) \* y1;

double project\_y1 = rotation(0,1) \* x1 + rotation(1,1) \* y1;

project\_x1 = scale \* project\_x1 \* bounding\_box.width / 2.0;

project\_y1 = scale \* project\_y1 \* bounding\_box.height / 2.0;

int real\_x1 = (int)project\_x1 + landmark\_x;

int real\_y1 = (int)project\_y1 + landmark\_y;

real\_x1 = max(0,min(real\_x1,image.cols-1));

real\_y1 = max(0,min(real\_y1,image.rows-1));

/\*fixed bug, by Ming, 2015.08.20

double project\_x2 = rotation(0,0) \* x2 + rotation(0,1) \* y2;

double project\_y2 = rotation(1,0) \* x2 + rotation(1,1) \* y2;

\*/

double project\_x2 = rotation(0,0) \* x2 + rotation(1,0) \* y2;

double project\_y2 = rotation(0,1) \* x2 + rotation(1,1) \* y2;

project\_x2 = scale \* project\_x2 \* bounding\_box.width / 2.0;

project\_y2 = scale \* project\_y2 \* bounding\_box.height / 2.0;

int real\_x2 = (int)(project\_x2 + landmark\_x);

int real\_y2 = (int)(project\_y2 + landmark\_y);

real\_x2 = max(0,min(real\_x2,image.cols-1));

real\_y2 = max(0,min(real\_y2,image.rows-1));

int pdf = (int)(image(real\_y1,real\_x1))-(int)(image(real\_y2,real\_x2));

if (pdf < rand\_forest[iter].nodes\_[currnode].thresh){

currnode =rand\_forest[iter].nodes\_[currnode].cnodes[0];

}

else{

currnode =rand\_forest[iter].nodes\_[currnode].cnodes[1];

bincode += pow(2, rand\_forest[iter].max\_depth\_-2-i);

}

}

## 三、LBFRegressor.cpp 函数GetCodefromTree

int LBFRegressor::GetCodefromTree(const Tree& tree,

const Mat\_<uchar>& image,

const Mat\_<double>& shape,

const BoundingBox& bounding\_box,

const Mat\_<double>& rotation,

const double scale){

int currnode = 0;

int bincode = 1;

for(int i = 0;i<tree.max\_depth\_-1;i++){

double x1 = tree.nodes\_[currnode].feat[0];

double y1 = tree.nodes\_[currnode].feat[1];

double x2 = tree.nodes\_[currnode].feat[2];

double y2 = tree.nodes\_[currnode].feat[3];

/\*fixed bug, by Ming, 2015.08.20

double project\_x1 = rotation(0,0) \* x1 + rotation(0,1) \* y1;

double project\_y1 = rotation(1,0) \* x1 + rotation(1,1) \* y1;

\*/

double project\_x1 = rotation(0,0) \* x1 + rotation(0,1) \* y1;

double project\_y1 = rotation(1,0) \* x1 + rotation(1,1) \* y1;

project\_x1 = scale \* project\_x1 \* bounding\_box.width / 2.0;

project\_y1 = scale \* project\_y1 \* bounding\_box.height / 2.0;

int real\_x1 = project\_x1 + shape(tree.landmarkID\_,0);

int real\_y1 = project\_y1 + shape(tree.landmarkID\_,1);

real\_x1 = max(0,min(real\_x1,image.cols-1));

real\_y1 = max(0,min(real\_y1,image.rows-1));

/\*fixed bug, by Ming, 2015.08.20

double project\_x2 = rotation(0,0) \* x2 + rotation(0,1) \* y2;

double project\_y2 = rotation(1,0) \* x2 + rotation(1,1) \* y2;

\*/

double project\_x2 = rotation(0,0) \* x2 + rotation(1,0) \* y2;

double project\_y2 = rotation(0,1) \* x2 + rotation(1,1) \* y2;

project\_x2 = scale \* project\_x2 \* bounding\_box.width / 2.0;

project\_y2 = scale \* project\_y2 \* bounding\_box.height / 2.0;

int real\_x2 = project\_x2 + shape(tree.landmarkID\_,0);

int real\_y2 = project\_y2 + shape(tree.landmarkID\_,1);

real\_x2 = max(0,min(real\_x2,image.cols-1));

real\_y2 = max(0,min(real\_y2,image.rows-1));

int pdf = (int)(image(real\_y1,real\_x1))-(int)(image(real\_y2,real\_x2));

if (pdf < tree.nodes\_[currnode].thresh){

currnode =tree.nodes\_[currnode].cnodes[0];

}

else{

currnode =tree.nodes\_[currnode].cnodes[1];

bincode += pow(2, tree.max\_depth\_-2-i);

}

}

return bincode;

};