Problem: Five output color images corresponding to the input images need to be created.

Image 1:

```
   double PI = 3.141592654;                    //pi constant
   double amplitude = 2.5;                 //amplitude constant for sine wave
   double frequency = 3.0;                 //frequency constant for sine wave
   double  phase_x = 0.0, phase_y = 0.0;     //phase constants for sine wave
   int  xx, yy;
r = (int) amplitude*(float)((double)x*sin(1.0*PI*frequency*(double)t/(double)H_COL - phase_x)
+ (double)y*cosh(0.0*PI*frequency*(double)t/(double)H_COL - phase_y)) + 300;
xx = r * sin(2.0*PI*(double)t/(double)H_COL - phase_x);
yy = r * cos(2.0*PI*(double)t/(double)H_ROW - phase_y);
if (xx >= -H_COL/2 && xx < H_COL/2 && yy >= -H_ROW/2 && yy < H_ROW/2)
        h[xx+H_COL/2][yy+H_ROW/2] = h[xx+H_COL/2][yy+H_ROW/2] + 1;
```

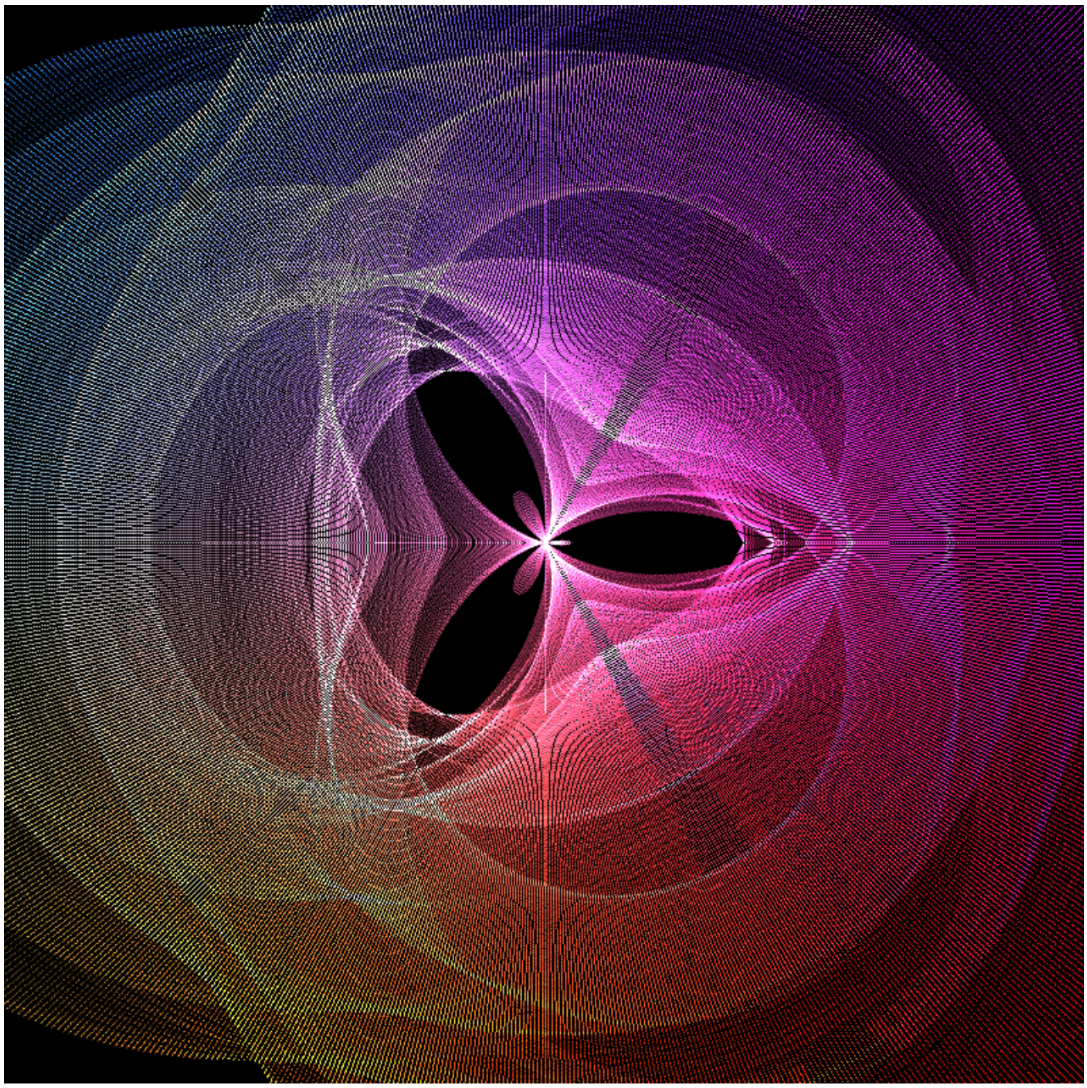

Image 2:

```
double PI = 3.141592654;                    //pi constant
```

```
   double amplitude = 2.5;              //amplitude constant for sine wave
   double frequency = 3.0;
r = (int) amplitude*(float)((double)x*tan(1.0*PI*frequency*(double)t/(double)H_COL - phase_x)
+                          (double)y*cosh(0.0*PI*frequency*(double)t/(double)H_COL - phase_y)) +
250;
                /* Creating Circular Hough Space */
                        xx = r * sin(2.0*PI*(double)t/(double)H_COL - phase_x);
                yy = r * cos(2.0*PI*(double)t/(double)H_ROW - phase_y);
                if (xx >= -H_COL/2 && xx < H_COL/2 && yy >= -H_ROW/2 && yy < H_ROW/2)
```

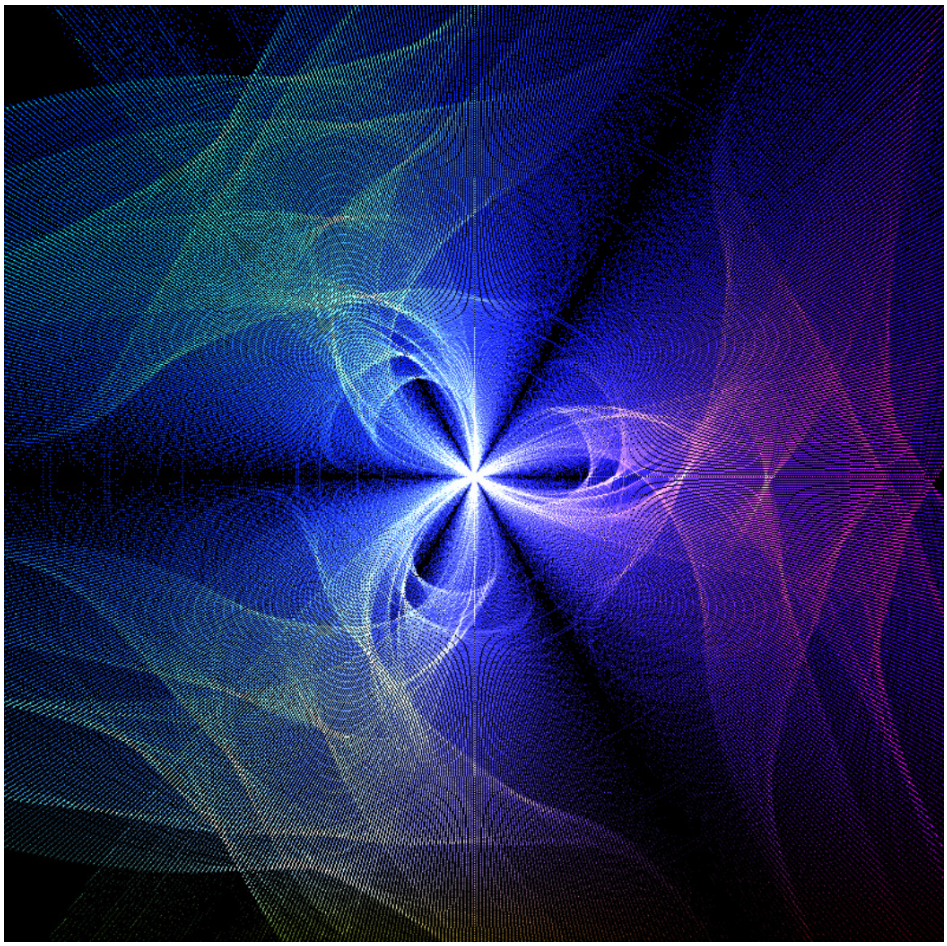

Image 3:

```
double amplitude = 2.5;              //amplitude constant for sine wave
   double frequency = 3.0;           //frequency constant for sine wave
   double  phase_x = 0.0, phase_y = 0.0
r = (int) amplitude*(float)((double)x*tan(5.0*PI*frequency*(double)t/(double)H_COL - phase_x)
+                          (double)y*sinh(0.0*PI*frequency*(double)t/(double)H_COL - phase_y)) +
250;
                /* Creating Circular Hough Space */
                        xx = r * sin(2.0*PI*(double)t/(double)H_COL - phase_x);
```

```
yy = r * cos(2.0*PI*(double)t/(double)H_ROW - phase_y);
if (xx >= -H_COL/2 && xx < H_COL/2 && yy >= -H_ROW/2 && yy < H_ROW/2)
```

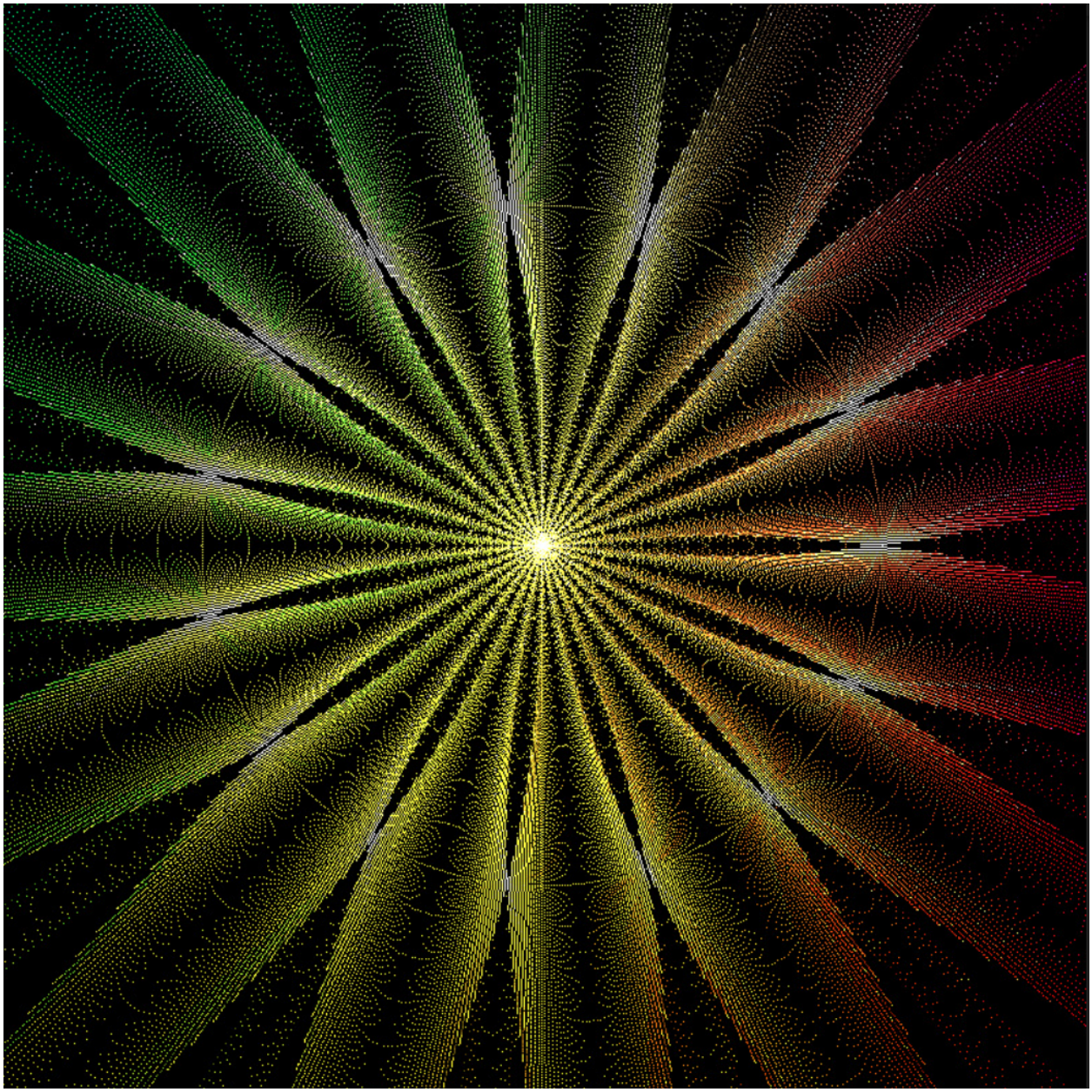

Image 4:

```
double PI = 3.141592654;                    //pi constant
   double amplitude = 5.0;              //amplitude constant for sine wave
   double frequency = 2.0;              //frequency constant for sine wave
   double  phase_x = 0.0, phase_y = 0.0;
r = (int) amplitude*(float)((double)x*tgamma(2.0*PI*frequency*(double)t/(double)H_COL -
phase_x)  +                    (double)y*lgamma(2.0*PI*frequency*(double)t/(double)H_COL -
phase_y));
```

```
/* Creating Circular Hough Space */
        xx = r * tgamma(2.0*PI*(double)t/(double)H_COL - phase_x);
yy = r * lgamma(2.0*PI*(double)t/(double)H_ROW - phase_y);
if (xx >= -H_COL/2 && xx < H_COL/2 && yy >= -H_ROW/2 && yy < H_ROW/2)
   h[xx+H_COL/2][yy+H_ROW/2] = h[xx+H_COL/2][yy+H_ROW/2] + 3
```

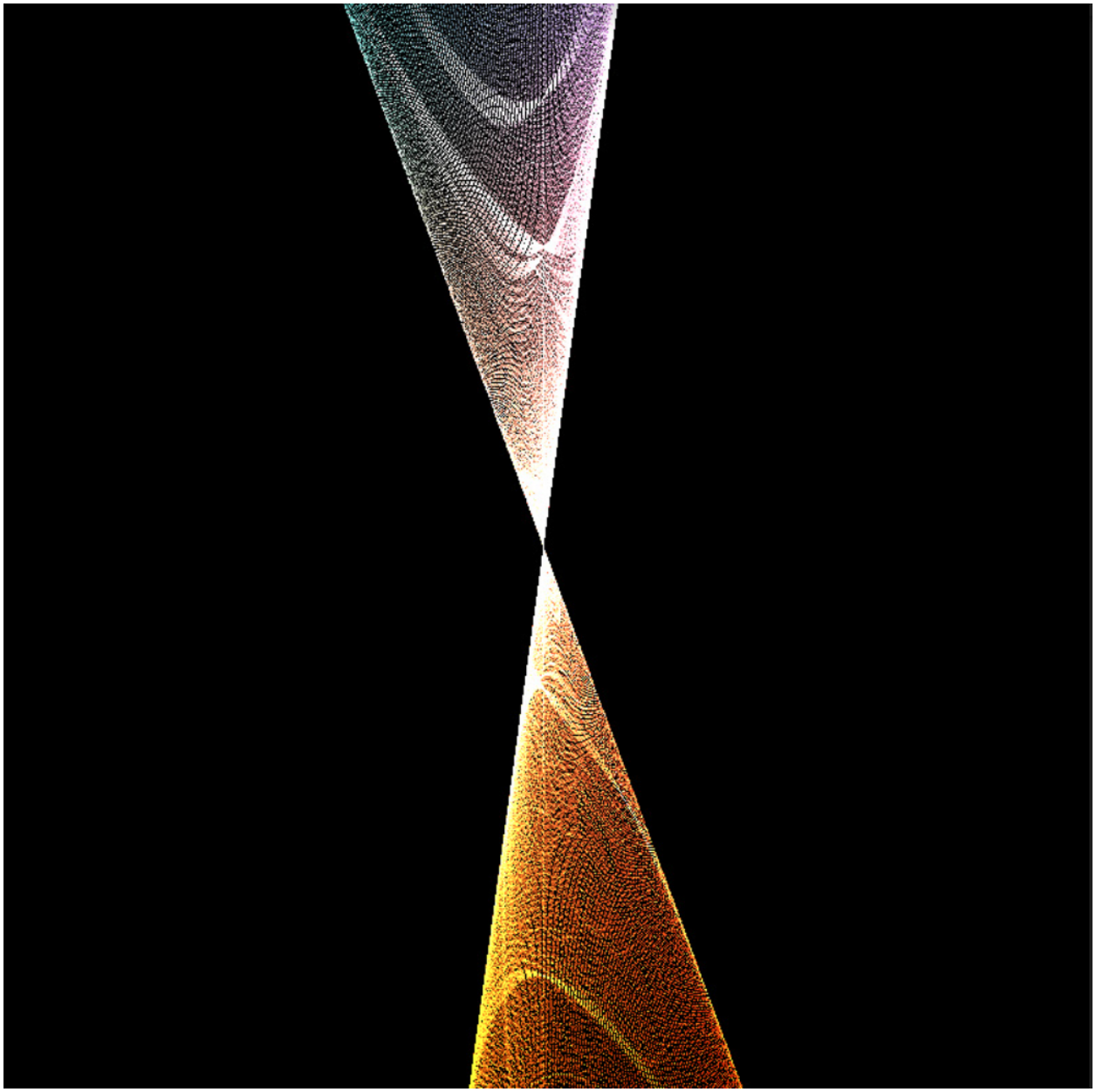

Image 5:

```
double amplitude = 2.5;                 //amplitude constant for sine wave
   double frequency = 3.0;              //frequency constant for sine wave
   double  phase_x = 0.0, phase_y = 0.0;
```

```
r = (int) amplitude*(float)((double)x*cos(100.05*PI*frequency*(double)t/(double)H_COL -
phase_x)  +                    (double)y*sinh(3.0*PI*frequency*(double)t/(double)H_COL -
phase_y)) + 300;
                /* Creating Circular Hough Space */
                        xx = r * sin(3.0*PI*(double)t/(double)H_COL - phase_x);
                yy = r * cos(4.0*PI*(double)t/(double)H_ROW - phase_y);
                if (xx >= -H_COL/2 && xx < H_COL/2 && yy >= -H_ROW/2 && yy < H_ROW/2)
                   h[xx+H_COL/2][yy+H_ROW/2] = h[xx+H_COL/2][yy+H_ROW/2] + 1;
```