where the magnitude |R (0,0)| is retained, and then, taking the IFFT of the new spectrum magnitude and leaving the phase of the repair window R unchanged.

A scratch introduces high magnitude values in the frequency spectrum of the image at issue. In most images, the spectrum of the sample window is useful only as a starting point for determining/generating data for repairing(repair data) the scratch area. Using the magnitude of the sample window frequency spectrum to replace repair window frequency spectrum as repair data would cause the information in the known, or non-scratched, portions of the repair window to be lost. Such a constraint would not represent a convex set as required by the present invention. In view of this, the minimum of the values between either the sample window or repair window values for each frequency within the frequency spectrum of the repair window, will be used as the new image data values. In this way, the effect of the scratch area can be reduced without losing all repair window frequency spectrum data. As a result degradation in the quality of non-scratched areas of the repair window is avoided. The d.c. component is retained since, taking the minimum magnitude value at d.c. can cause the overall brightness of the image to be reduced. C_1 is closed and convex. Note that P_1 is a projection because it makes the minimum change possible to make its input satisfy C_1. Constraint #2 is defined as:

$$C_2=\{r:r(j,k)\in\Im,\ 0\leqq r(j,k)\leqq 255\} \qquad \text{Equation 4}$$

where j and k represent column and row number of the pixels in the image.

Thus, C_2 is the set of all matrices which are real and have pixel values that are between 0 and 255, the typical range of luminance values in a 256 gray scale. To define projection #2, $P_2(q)$, it may be assumed that q(j,k)=a(j,k)+ib(j, k), i.e. q is a complex array indexed by j,k. Then:

$$P_2(q(j,k)) = \begin{cases} a(j,k) & \text{if } 0 \le a(j,k) \le 255 \\ 0 & \text{if } a(j,k) < 0 \\ 255 & \text{if } a(j,k) > 255 \end{cases} \qquad \text{Equation 5}$$

Since the output of P_1 may not be real and the real part may be outside [0, 255], P_2 converts its input to a real array and hence an image.

Where W is the set of coordinate pairs where the scratch or wire mask is 1 (W is the set of pixel locations which comprise the scratch/damaged area of the image), then, constraint #3 is defined as:

$$C_3=\{r:r(j,k)=r_0(j,k),\ (j,k)\notin W\} \qquad \text{Equation 6}$$

This defines a set of complex matrices which are identical to the original repair window area exclusive of the actual scratch/damaged pixel area. Let w be the binary mask which is 1 at any pixel location which comprises a part of the defined scratch/damage area and 0 at all other pixel locations. Then projection #3 (Step **11**) is defined as follows:

$$P_3(r)=rw+r_0(1-w) \qquad \text{Equation 7}$$

This step provides for maintaining non-scratched/non-damaged pixels of the original image, since there is no point to needlessly replacing good image data.

It will be noted that C_1, C_2 and C_3 are closed convex sets. Also P_2 and P_3 are projections since like P_1, they have been defined in such a way that they make the least amount of change in the input data necessary to make the input data confirm with the specified corresponding constraint(s).

FIG. **5** is diagram showing the relation between the projections P_1, P_2& P_3 and respective steps illustrated in the flowchart of FIG. **2**. It can be seen that projection P_1 relates to steps **6–9**, while projection P_2 relates to step **10**. Projection P_3 relates to step **11**.

With reference to FIG. **2**, FIG. **6** and FIG. **7**, a further embodiment of the present invention will be described. This embodiment allows for the definition of a sample window **65** and repair window **70** via "brush" like strokes, or movements of a cursor, or brush tip, **60** across the scratch area of the image at issue which is displayed on the display device **10**.

In this implementation a scratch area **40** of an image **50** is defined through movement of a cursor, or brush, **60** across the scratch area **40** the image **50**, displayed on the display device **10**. Once the scratch area **40** has been defined, binary mask data is generated which distinguishes the defined scratch/damage area of the image from the remaining areas of the image. This binary mask data is then stored into memory **25**. The cursor/brush **60** is generated and displayed in accordance with controller **30** based upon input from a user input device. User input device may be, for example, a mouse **15**, a trackball, joystick, pen or pen and tablet **20**, or the like. Once the scratch area **40** has been defined and binary mask data generated and stored in memory, the operator can move the cursor **60** on the display device as desired from a starting point **62** outside the defined scratch area **40**, inward toward the scratch area **40**. The position and movement of the cursor **60** on the image/display device is monitored by controller **30** as it moves from the start point outside the defined scratch area **40** toward the scratch area **40**. Once the cursor **60** reaches the outer boundary of the defined scratch area **40** (point **80**), controller **30** causes a rectangular sample window **65** to be defined. This rectangular sample window **65** is, for example, as long as the distance between the start point and the point **80** on the outer boundary of the scratch area **40**. The width of the rectangular sample window **65** may be predefined to be, for example, the width of the cursor **60**. It may also be defined to be a predetermined number of pixels wider or narrower than the cursor **60**. It may also be adjustable to meet the specific needs of the user/operator.

The repair window **70** is then defined. Since the present invention requires the dimensions of the sample window **65** and the repair window **70** to be of the same size and dimensions, controller **30** causes a repair window **70** area to be generated which abides by this size restriction. The alignment of repair window **70** is defined by moving cursor **60** to a second point **75** outside the scratch area **40** and outside the sample window area **65**. The repair window **70** is generated, for example, so as to run length wise from the second point **75** back toward the point **62**, along the line **85** defined thereby.

Once the sample window **65** and repair window **70** are defined, the user may utilize the cursor **60** to virtually paint, or brush, replacement pixel data into/onto the repair area **70**. Replacement pixel data is generated in accordance with the above described process and calculations. Each stroke of the cursor **60** could be viewed as the equivalent of one iteration of the process of generating appropriate replacement data. Of course any number of iterations of the process could be carried out with each stroke. This would be contingent upon how many iterations are assigned to each stroke.

The description herein of the present invention has assumed an image and display resolution of 16 bits. It will be recognized that the present invention can be extended to any image/display resolution, including, but not limited to,