document frame is received, it is stored until three frames are available. These three frames are combined by a voting method to create a first order frame combination. The voting method creates a first order frame wherein the color of each pixel in the first order frame is determined by pixels in corresponding locations in the three received frames, the pixel being dark if two or three of the corresponding pixels are dark, otherwise the pixel is fight. Of course, other numbers of frames can be used and other voting thresholds can be used.

The first order frame is stored, and when three first order frames are available, they are similarly combined into a second order frame. It should be apparent that the number of dark pixels in a frame goes down as the order of the frames goes up, when there are fewer dark pixels than light pixels in the frames. Thus, with compression, less memory is required to store higher order frames, and the stored frames and higher order frames can be stored in a fraction of the space required by even a binary frame buffer, particularly if a compression technique such as LZW is employed.

FIG. 9 is a block diagram of an apparatus designed to accumulate these frames. Accumulator system **300** accepts digital frames **301** and outputs a defect list **306** which indicates both fight and dark defects. System **300** includes an accumulator **302**, a compressor **304**, dark defect frame storage **308**, and light defect frame storage **310**. It should be understood that the roles of dark and light pixels are reversed in the figure as dark pixels are more prevalent in the frames than white pixels.

Accumulator **302** is coupled to compressor **304**, and compressor **304** is coupled to dark defect frame storage **308**, and light defect frame storage **310**. In operation, accumulator **302** accepts frames, requests stored frames from compressor **304**, and sends frames to compressor **304** for storage in one of the storage areas. When compressor **304** is retrieving a frame it acts as a decompressor, and when storing it acts as a compressor. This way, accumulator **302** deals with uncompressed frames while the frames are always stored in compressed form. Compressor **304** decompresses dark defect frames from, and compresses the frames to, storage **308**, and also decompresses light defect frames from, and compresses the frames to, storage **310**.

In the preferred embodiment, the accumulation and compression/decompression is done on a row-by-row basis with the current frame and the stored frames, by decompressing, adding, and recompressing scan lines separately, so that memory need not be allocated for an entire uncompressed frame. The number of orders of frames stored is variable depending on application requirements, although FIG. 9 shows three orders of documents. Once a number of documents have been scanned and a sufficiently high order frame is created, that frame is output as the defect list.

Using this method, the contents of the documents quickly fades from the higher order frames, leaving only the added defects. For example, with frames being combined at each order in threes and the threshold being two out of three, 81 frames are combined to form a fourth order frame. For a pixel in a fourth order frame to be a dark pixel, dark pixels would have to have been present at that pixel's location in at least sixteen of the first order frames. Assuming there are no defects and the frames are independently populated with 10% dark pixels and 90% light pixels, the fourth order frame would contain less than one dark pixel for every 10^{14} pixels. Therefore, the dark pixels present in a fourth order frame can be assumed to be defects.

White defects can be detected even more simply. By logically "OR"ing all the images, white defects can be detected. The frame stored in light defect frame is initially an all white image. As each image is provided to accumulator **302**, it is "OR"ed pixel-by-pixel with the light defect frame. As images are provided, the frame becomes increasingly populated with dark pixels, except where each image includes similar margins and where white defects exist. After a sufficient number of images have been processed by accumulator **302**, in general, only white defects are left inside the margins. Accumulator **302** detects the white margins of the images by locating the largest rectangle whose perimeter comprises at least 99.9% dark pixels. Within this rectangle, white pixels are identified as defective, and are output as part of defect list **306**.

The above description is illustrative and not restrictive. Many variations of the invention will become apparent to those of skill in the art upon review of this disclosure. For example, this invention could be implemented by dedicated hardware, a programmed digital computer, or both. The scope of the invention should, therefore, be determined not with reference to the above description, but instead should be determined with reference to the appended claims along with their full scope of equivalents.

We claim:

1. A method for automatically identifying a set of defect locations in a digital image where a defect location in the set results from a defect in a digitizing means used to generate the digital image from an image of a document, wherein the digital image comprises a plurality of pixel locations with a pixel color value for each pixel location, the method comprising the steps of:

scanning multiple documents to form a plurality of digital images;

assigning each pixel location of each digital image of the plurality of digital images a pixel color value, wherein a pixel is characterized by a location and a color;

accumulating, for each pixel location in a region of interest, a count of documents in which said each pixel location contains a pixel of a common color value; and

adding pixel locations which have a count higher than a threshold to the set of defect locations.

2. The method of claim 1, further comprising the step of removing particular pixel locations from the set of defect locations when said particular pixel locations are associated with a pixel pattern known to be present on documents of said multiple documents scanned in said scanning step.

3. The method of claim 1, wherein the common color value is a color value associated with either the color black or the color white.

4. The method of claim 1, wherein the region of interest is an entire image.

5. The method of claim 1, wherein the region of interest is limited by an mount of memory available in said accumulating step.

6. The method of claim 1, wherein the region of interest is a column which varies over the possible pixel locations of a document.

7. The method of claim 1, wherein said accumulating step is only performed if pixel locations in consecutive documents have pixels of a common color value in a pixel location, and further comprising the step of removing pixel locations for which pixels in consecutive documents are of different colors.

8. A method for automatically identifying a set of defect locations in a digital image where a defect location in the set