RedBot
- a tool for improving red-eye correction

Robert Ulichney, Matthew Gaubatz, JM Van Thong
Hewlett-Packard Company
Cambridge, MA U.S.A.

Abstract

Red-eye is the appearance of an unnatural red hue in the pupils of a flash photography subject. The redness is caused by the reflection of the flash off blood vessels of the subject's retina. This problem has long bothered camera users, and is especially common with compact cameras due to their inherently small lens-to-flash angles. We have developed completely automatic algorithms for detecting and correcting red-eye. The lion's share of the problem is the detection phase. One solution employs a robust human face detector based on a cascade of multi-scale classifiers. In a detected face, a mask of red-eye pixels is created and successively refined by combining metrics that detect redness, color variation and size. Once the offending red eyes are located, a simple tapered chrominance desaturator combined with some luminance attenuation completes the correction.

One objective measure used to quantify the success of a red-eye correction algorithm plots the percent of corrected red-eyes versus percent of false positives pixels. A false positive, that is, the removal of red in regions that are not red-eyes, is considered a severe error, often more offensive than the original red-eye. The success and quality of many image enhancement algorithms, specifically those that address a particular image defect such as red-eye, depend on the breadth of test cases available to measure performance. The need for such images motivates the creation of RedBot, a web service for automatic correction of photo flash induced red-eye. Users enjoy free correction of their red-eye photos in exchange for allowing the use of submitted "real-world" image for testing purposes. Upon the return of the corrected image, the user is invited to complete a short feedback section. The goal is to collect and annotate over ten thousand diverse red-eye photos.

RedBot is implemented on the Microsoft .NET framework. Submitted images are uploaded to the RedBot web site and processed within a synchronous call. It usually takes less than a second to remove red eye artifacts in an average size image. To achieve scalability, the system is distributed across several servers loosely connected via web services. Periodic logging allows the collection of submitted images in a central repository.

Algorithm enhancement depends on refinement based on known failures. One important benefit from such a large database is the exposure of many new "problem" images for making solutions more robust. Another benefit is the ability to more accurately quantify the success of candidate algorithms or algorithm changes. In order to automate algorithm testing, however, it is necessary to annotate the submitted images, i.e. specify the location and nature of the red-eye artifacts. A first pass of this task is accomplished by verifying the success of a candidate algorithm on an image. Therefore, the user feedback, which provides a quantified measure of quality along with one count of false positives and false negatives, is particularly valuable in this highly subjective field of image enhancement.

We will launch RedBot in late October, 2003 at URL
http://hpl.hp.com/research/redbot/.