

적목현상 특징을 이용한 적목현상 보정 알고리즘 및 하드웨어 구현

Algorithm and Hardware Implementation of Redeye Correction Using the Redeye Features

이성진*, 장원우**, 최원태***, 김석찬****, 강봉순*****

Chris Songjin Lee*, Wonwoo Jang**, Wontae Choi***, Sukchan Kim****, Bongsoon Kang*****

요약

본 논문은 적목현상 보정을 위한 알고리즘에 관한 것이다. 자연스럽게 보정을 위해 적목현상이 발생한 이미지에서 적목현상이 발생한 위치를 추정하고 적목현상의 정도를 판단하여 적절한 보정을 적용적으로 적용할 수 있도록 하였다. 붉은 색, 피부색, 반사광색을 이용하여 적목현상이 발생했을 가능성이 있는 픽셀을 추출하고 픽셀들을 그룹화하여 레이블링 하였다. 레이블링 된 각 그룹들 중에서 그룹의 가로와 세로의 비율, 크기, 밀도, 흰자위색과 피부색등을 이용하여 어떠한 그룹이 적목현상이 발생한 그룹인지를 판단하였다. 적목현상이 발생한 위치는 블러링 효과를 만들어 자연스럽게 보정하였다. 제안된 알고리즘을 하드웨어로의 설계할 경우 하드웨어 자원의 효율성을 극대화하기 위해 알고리즘을 최소한의 메모리를 사용하여 개발하였다.

Abstract

In this paper, we proposed an algorithm of redeye correction. For naturally redeye correction, we assumed positions of the redeye at an image which produced redeye, and we estimated rate of the redeye to apply the appropriate redeye correction suitably. We extract and label pixels those are possible of generating redeye using red, skin and reflected light color. The each labeled group is decided by rates of length and width, dimension, density, the color of white of the eye and reflected light color of groups for the redeye group. We corrected positions of redeye using blurring effect, naturally. In the case of designing the proposed algorithm, we designed the redeye correction hardware using the minimum of memories for efficiency of hardware.

Keywords : Redeye, Mask, Grouping, Correction

I. 서론

CIS(Complementary metal-oxide semiconductor Image Sensor)와 CCD(Charge Coupled Device)같은 이미지 센서의 보급으로 디지털 카메라 및 휴대 전화와 같은 Portable Device가 대중화되었고, 고성능의 영상 장치를 선호하게 되었다[1]. 따라서 이미지 센서로부터 자연스럽게 깨끗한 영상을 촬영하기 위하여 밝기조정, 선명도 향상, 색도조정 등과 같은 더욱 향상된 영상신호처리 기법이 개발

되고 있다. 적목현상 보정은 이러한 영상신호처리 기법 중 하나이다.

적목현상은 어두운 곳에서 동공이 확장되어 있을 때 플래시를 사용하여 촬영할 경우, 망막의 혈관이 촬영되는 현상이다[2]. 적목현상을 해결하기 위한 기존의 방법 중, 물리적인 해결 방법으로는 렌즈와 플래시 간 거리를 멀리하거나[3] 플래시를 여러 번 작동시키는 방법[4]이 있고, 디지털적인 방법으로는 특징 기반 접근법[5], Fuzzy Theory[6] 등의 연산을 수행하는 방법이 있다. 물리적인 방법의 경우 모바일 기기의 특징인 소형화에 제한을 받고, 디지털적인 방법은 복잡한 연산을 수행함으로써 모바일 기기와 같은 소형 장비에 사용되는 마이크로 컨트롤러의 성능에 제한적이다[7].

본 논문에서는 자동적으로 적목현상의 위치를 추정하고 색을 보정하도록 하였으며, 영상을 저장하지 않고 실시간으로 ASIC을 이용하여 처리하였다. 적목현상이 발생한 위치를 추정하기 위하여 수백 개의 적목현상 발생 이미지로부터 적목의 정보, 플래시에 의한 반사광과 피부 색 정보를

*동아대학교 전자공학과 석사과정

**동아대학교 전자공학과 박사과정

*** (주) 삼성전기 중앙연구소 수석연구원

****부산대학교 전자공학과 부교수

*****교신저자 : 동아대학교 전자공학과 부교수

투고 일자 : 2009. 2. 4 수정완료일자 : 2009. 7. 21

계재확정일자 : 2009. 7. 29

※ 이 논문은 동아대학교 학술연구비 지원에 의하여 연구되었음.

분리하여 연구하였고, 적목이 띄는 색 정보를 찾고 보정하기 위해 RGB, YCbCr 색 좌표계[8]를 이용하였다. 검출된 결과는 그룹화를 통하여 좌표로 인지하게 되고, 해당 좌표의 붉은색을 유지가 원하는 색으로 변환하며 블러링 효과를 이용하여 더욱 자연스럽게 보정한다.

II. 적목현상 보정을 위해 제안된 알고리즘

그림 1은 본 논문에서 제안한 적목현상 보정의 전체 흐름을 도시하였다. 적목현상이 발생한 영상이 입력되면, 우선 적목현상이 발생한 픽셀을 검출하기 위하여 Red, Skin, Highlight, White의 4개의 마스크를 추출한다. 이 4개의 마스크 중 Red, Skin, Highlight Mask는 합성하여 그룹화 과정의 입력으로 사용된다. 그리고 그룹화된 결과에 White Mask를 사용하여 최종적으로 적목을 추출한다. 마지막으로 추출된 적목을 보정 단계에서 보정함으로써 영상에서 적목을 제거하게 하였다.



그림 1. 적목현상 보정의 전체 흐름도
Fig. 1. Flowchart of redeye correction

2.1 적목현상 발생 픽셀 추출을 위한 Mask

그림 2는 원 영상에서 적목현상의 좌표를 추정하기 위해 Red, Highlight, Skin, White Mask를 추출하는 과정을 도시하였다. Red, Highlight, Skin Mask는 5X5 morphology[9] 연산으로 추출된 영역을 확대하여 사용하였다[10]. 각각의 마스크 추출을 위해 RGB와 YCbCr 색 좌표계를 사용하였다.

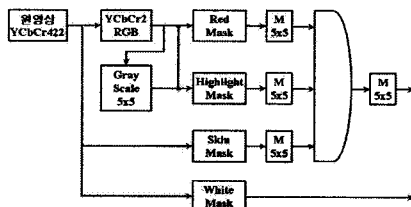


그림 2. 입력 영상으로부터 마스크 추출과정
Fig. 2. Extraction process of mask from input image

그림 3은 적목현상이 발생한 이미지를 사용하여 그림 2의 마스크 추출 결과를 보여주고 있다. 그림 3의 (a)는 적목현상 보정을 위한 예시로서 암실에서 촬영하여 양쪽 눈 모두 적목 현상이 발생한 영상이고 나머지 그림들은 (a)를 입력으로 마스크를 추출한 결과이다. 그림 3의 (b)는 Red Mask, (c)는 Skin Mask, (d)는 그레이 스케일 팽창 결과, (e)는 Highlight Mask, (f)는 White Mask를 보여준다. 이러한 마스크들의 추출 방법은 아래에 간략하게 설명하겠다. 그리고 마지막으로 (g)는 Red, Skin, Highlight Mask의 합성 결과로써 그룹화 과정의 입력으로 사용된다.

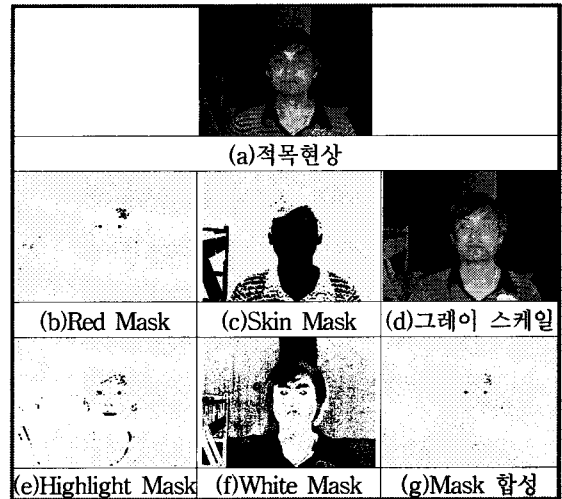


그림 3. 마스크 추출 결과

Fig. 3. Extraction process of mask from input image

$$\begin{aligned}
 red_1 \times R &> G & (1) \\
 red_2 \times R &> B & (2) \\
 red_3 \times R &> R + G + B & (3) \\
 red_3 \times G &> R + G + B & (4) \\
 red_3 \times B &> R + G + B & (5) \\
 R + G + B &< red_4 & (6) \\
 R + G + B &> red_5 & (7)
 \end{aligned}$$

식 (1)~(7)은 Red Mask를 검출하는 데 사용되는 수식으로써, 적목현상이 발생한 이미지의 적목에 해당하는 픽셀을 분석하여 구하였다. 식 (1), (2)는 각각 R과 G, R과 B의 비율($red_1=0.625$, $red_2=0.625$)을 지정하는 수식, 식 (3)~(5)는 명도와 RGB 각각의 비율($red_3=5.818$)을 지정하는 수식, 식 (6), (7)은 명도의 허용 범위($red_4=612$, $red_5=76$)를 지정하는 수식이다. 그리고 Skin Mask는 YCbCr 색좌표계에서 피부영역에 해당하는 CbCr 정보를 이용하여 추출하였다[11].

$$\begin{aligned}
 R &> high_1 & (8) \\
 G &> high_2 & (9) \\
 B &> high_3 & (10) \\
 high_4 \times R &< R + G + B & (11) \\
 high_4 \times G &< R + G + B & (12) \\
 high_4 \times B &< R + G + B & (13) \\
 high_5 \times R &> G & (14) \\
 high_6 \times R &> B & (15)
 \end{aligned}$$

그레이 스케일 팽창 결과는 Highlight Mask를 추출하는데 사용되며, 5X5 픽셀 내에서 밝은 픽셀의 값을 타겟 픽셀에 적용하는 그레이 스케일 팽창기법을 사용하여 구하였다[12]. 그레이 스케일 팽창기법은 플래시의 반사광(Highlight 영역)을 강조하기 위한 방법이다. 그리고 일반적으로 플래시의 반사광은 수정체에 반사되기 때문에 이를 이용하여 적목현상이 발생하는 위치를 추정할 수 있다. 식 (8)~(15)는 Highlight Mask를 검출하는 데 사용되는 수식으로써, 적목현상 이미지의 광원이 수정체에 반사되는 픽셀을 분석하여 구하였다. 식 (8)~(10)은 Highlight Mask에서 RGB의 범위($high_1=150$, $high_2=115$, $high_3=115$)를 지정하는 수식, 식

(11)~(13)은 명도와 RGB 비율($high_4=4.571$)을 지정하는 수식, 식 (14)와 (15)는 R과 G, R과 B의 비율($high_5=0.625$, $high_6=0.625$)을 지정하는 수식이다.

식 (16)~(20)은 White Mask를 검출하는 데 사용되는 수식으로써, 식 (16), (17)은 White Mask에서 명도의 범위($white_1=28$, $white_2=606$)를 지정하는 수식, 식 (18)~(20)은 명도에 대해 RGB 각각의 값의 범위($white_3=3.556$)를 지정하는 수식이다. 그리고 White Mask의 경우, 그룹화된 결과가 적목인지 노이즈인지를 판단하는데 사용된다.

식 (1)~(20)의 각 비율들은 적목현상이 빈번하게 발생할 수 있는 빛이 없는 매우 어두운 실내 환경에서 카메라의 플래시를 이용하여 촬영된 약 400개의 눈 이미지를 이용한 실험을 통하여 결정하였다. 그리고 피사체와 카메라의 거리는 약 1~1.5m 간격으로 이미지를 촬영하였다. 4개의 마스크는 빛의 변화에 민감하기 때문에 빛의 밝기에 따라 각 수식의 비율 값을 달리 적용해야 하고, 제안한 알고리즘에서는 이 값들을 외부에서 조절 할 수 있도록 하였다. 그리고 본 논문에서는 사람이 안경을 착용하였을 경우에 대한 고려는 배제하고 안경을 착용하지 않은 경우에 한하여 실험하였다.

- $R + G + B > white_1$ (16)
- $R + G + B < white_2$ (17)
- $white_3 \times R > R + G + B$ (18)
- $white_3 \times G > R + G + B$ (19)
- $white_3 \times B > R + G + B$ (20)

2.2 적목현상 발생 픽셀의 그룹화

2.2.1 그룹화 알고리즘

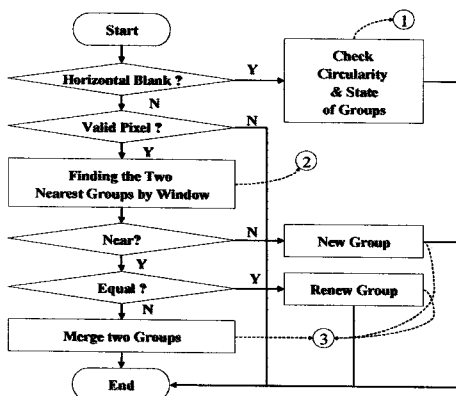


그림 4. 그룹화 알고리즘
Fig. 4. Labeling algorithm

그룹화 알고리즘은 그림 4와 같이 검색범위를 지정하여 그룹화하는 부분, 검색된 그룹의 유무에 따라 그룹을 병합하거나 갱신 또는 새롭게 생성 부분, 그룹화가 끝난 그룹의 특성을 판단하여 초기화하는 부분의 세부분으로 나눌 수 있다 [13]. 그룹화 알고리즘에 의해 생성되는 각 그룹의 정보는 영역내의 유효한 픽셀 수를 의미하는 grp_pixel_count , 그룹의 x방향과 y방향의 최솟값과 최댓값에 해당하는 x_min_pos , x_max_pos , y_min_pos , y_max_pos 로 구성된다.

본 논문에서는 최대 32개의 그룹을 생성할 수 있도록 데이터를 각각 32개씩 저장할 공간을 만들어 실험하였다. 그리고 한정된 하드웨어 공간을 효율적으로 활용하기 위해 눈일 가능성이 낮은 그룹의 데이터는 그룹핑 과정에서 즉시 초기화하여 하드웨어 공간을 절약하면서 최대한 유효한 그룹이 생성될 수 있도록 하였다. 또한 인접한 유효픽셀들은 하나의 그룹으로 지정하여 조밀하게 모여진 노이즈에 대처하여 더욱 효율성을 높였다.

식 (21)은 유효그룹을 판단하는 조건으로 위에서 아래의 순서대로 그룹의 유효픽셀 수(grp_pixel_count), 면적대비 유효픽셀의 비율($density=0.351$), 그룹의 가로(x)와 세로(y)의 비율에 대한 것이다. 식 (21)에서 $eye_cnt_max(=500)$ 와 $eye_cnt_min(=16)$ 은 각각 유효픽셀 수의 최댓값과 최솟값을 의미하고, $distance(=0.4)$ 와 $height(=0.5)$ 는 각각 그룹의 가로와 세로의 비율을 의미하며 식 (21)을 모두 만족하지 않으면 그룹은 초기화된다. 식 (21)의 각각의 비율 값들은 식(1)~(20)의 각 비율들을 설정한 방법과 마찬가지로 다양한 이미지를 이용한 실험을 통하여 결정하였으며, 피사체와 카메라의 거리에 따라 유효그룹을 판단하는 비율들에 변화가 필요하다. 본 논문에서는 약 1~1.5m 거리의 피사체를 촬영한 이미지에 대하여 비율을 결정하였으며, 피사체와의 거리가 달라질 경우를 고려하여 외부입력으로 식 (21)의 비율 값들을 조절할 수 있도록 하였다.

$$\begin{aligned}
 & grp_pixel_count < eye_cnt_max \\
 & grp_pixel_count > eye_cnt_min \\
 & density \times x \times y > grp_pixel_count \\
 & distance \times x > y \\
 & x < height \times y
 \end{aligned} \tag{21}$$

2.2.2 그룹화를 위한 검색 범위

표 1. 검색 영역 테이블(단위:pixel)
Table 1. Table of searching range

semil_index	1	2	3	4	5	6	7
scan_range	6	4	3	2	2	1	0

그룹화를 수행할 때, 패턴이 조밀한 경우 하나의 영역으로 판단하여 하드웨어 공간을 절약하기 위하여 표 1과 같은 검색범위를 설정하였다. 표 1은 검색영역 테이블로써 $semil_index$ 는 세로거리에 해당하고, $semil_index$ 에 해당하는 $scan_range$ 만큼 가로 방향으로 검색범위를 지정하였다.

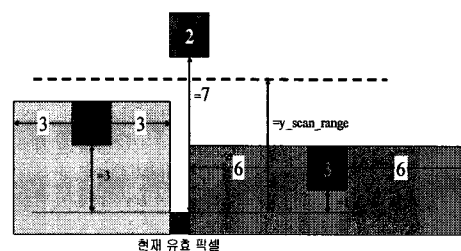


그림 5. 검색 실행[13]
Fig. 5. Searching

그림 5는 표 1의 검색 영역 테이블을 적용하여 검색하는 방법을 도시하고 있다. 1번 그룹의 경우 검색을 실시하는 픽셀과 세로 방향으로 3픽셀 떨어져 있다. 따라서 1번 그룹의 x_{min_pos} 와 x_{max_pos} 를 각각 3픽셀씩 확대하여 그 범위 안에 검색을 실시하는 픽셀이 있는지를 확인한다. 그 여부에 따라 검색을 실시하는 픽셀이 하나의 그룹으로 통합될지 아니면 새로운 그룹을 생성할지 결정한다. 그림 6은 그림 3의 (g)를 입력으로 그룹화를 수행한 결과이다.

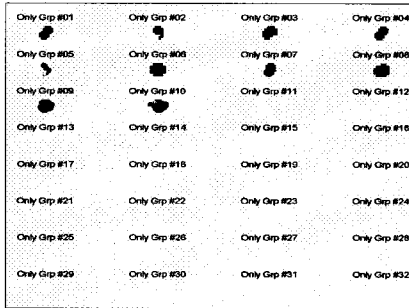


그림 6. 그룹화 결과
Fig. 6. The results of grouping

2.2.3 White Mask의 적용

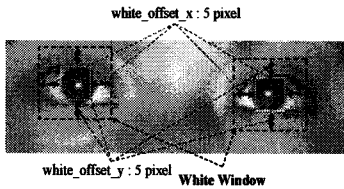


그림 7. 흰자위 범위
Fig. 7. Section of the white of the eye

그림 7은 눈동자에서 흰자위 범위(White Window)를 보여준다. 그룹화의 결과들 중에서 눈인지 아닌지 판단하기 위하여 각 그룹의 상하좌우 4방향으로 5픽셀(white_offset) 이내에 흰자위에 해당하는 영역이 존재하는지 확인하였다. 이는 카메라를 90도 회전시켜 촬영하는 경우를 고려한 것이다. 흰자위는 White Mask로 추출하였다.

그림 8은 그림 6에 대하여 양 옆에 흰자위에 해당하는 픽셀이 있는 경우를 검출하여 최종 그룹을 표시한 결과로써, 대부분의 눈이 아닌 그룹이 제거되었다.

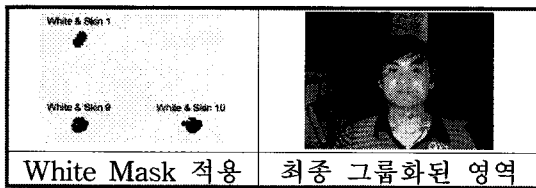


그림 8. 최종 그룹화된 적목
Fig. 8. The last group of redevye

2.3 적목현상 영역 보정

2.3.1 보정을 적용하는 영역

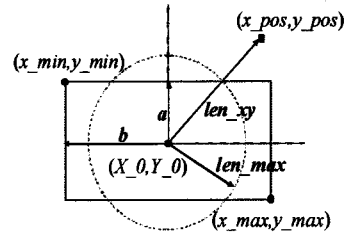


그림 9. 눈 보정 범위[7]
Fig. 9. Range of eye correction

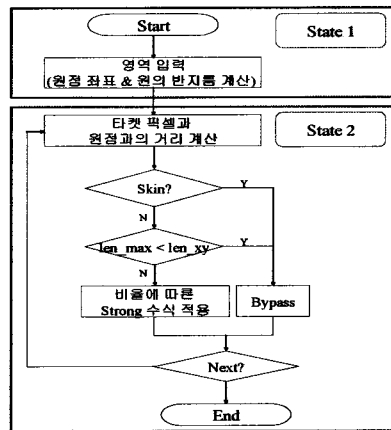


그림 10. 보정을 위한 알고리즘
Fig. 10. Algorithm for the correction

그림 9와 같이 중심점(X_0, Y_0)에서 보정할 픽셀(x_{pos}, y_{pos})의 거리(len_{xy})를 구하여 원의 반지름(len_{max})과의 비율에 따라 보정율을 다르게 적용하였다. 중앙에 가까울수록 보정율은 커지고, 원의 밖에 위치할 경우 고려하지 않는다. 그림 9에서 (x_{min}, y_{min})와 (x_{max}, y_{max})는 각각 설정된 그룹의 최소와 최대 좌표를 의미하며, a 와 b 는 각각 그룹의 세로와 가로 길이의 절반에 해당한다. a 와 b 를 이용하여 원의 반지름을 구하게 된다[7].

그림 10은 적목 보정을 위한 알고리즘이다. State 1은 초기화 단계로써, 중심점과 원의 반지름을 구한다. State 2는 입력된 픽셀과 중심점과의 거리를 계산하여, 반지름보다 작을 경우 색상을 보정하고 클 경우 고려하지 않는다. 색상을 보정할 경우 거리와 반지름의 비율이 커질수록 보정율은 커진다. 만약, 입력된 픽셀이 피부영역에 해당한다면 피부영역을 침범하지 않게 거리에 관계없이 보정하지 않는다.

2.3.2 보정을 위한 수식

$$Rate = \cos\left(\left(\frac{len_{xy}}{len_{max}} + 1\right) \times \pi\right) \times Gain + Offset \quad (22)$$

입력 YCbCr에 대하여, Y에는 명도의 보정율을 CbCr에는 색상의 보정율을 적용하여 Y'Cb'Cr'로 출력하는 식을 이용하여 보정하였다. 식 (22)는 명도의 보정율과 색상의 보정율에 적용되는 일반식으로 자연스러운 보정을 위하여

cosine형태의 곡선을 갖는다. 명도와 색상 모두 중심에 가까울수록 보정 정도가 크고 테두리에 가까울수록 원래의 값을 유지하게 하였다. 그리고 광원의 정도와 동공의 확장 정도에 따라 적목의 정도에 차이가 발생하기 때문에, 식 (22)의 Gain과 Offset을 적절히 조절하여 적목의 정도를 고려하여 더욱 자연스럽게 보정하게 하였다.

2.3.3 하드웨어 구현을 위한 보정 수치

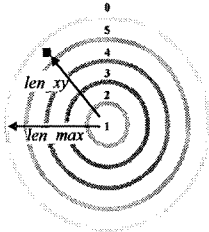


그림 11. 보정을 위한 동심원[14]
Fig. 11. Concentric circle for the Correction

표 2. 영역 구분을 위한 수식[14]

Table 2. Formulas for section classification

영역	영역구분	영역구분 단순화
1	$len_xy < \frac{1}{5} \times len_max$	$25 \times Len_xy < Len_max$
2	$len_xy < \frac{2}{5} \times len_max$	$25 \times Len_xy < 4 \times Len_max$
3	$len_xy < \frac{3}{5} \times len_max$	$25 \times Len_xy < 9 \times Len_max$
4	$len_xy < \frac{4}{5} \times len_max$	$25 \times Len_xy < 16 \times Len_max$
5	$len_xy < \frac{5}{5} \times len_max$	$Len_xy < Len_max$

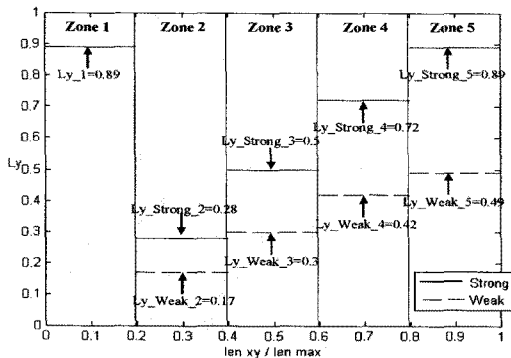


그림 12. 명도 보정을 위한 수치
Fig. 12. Values for luminance correction

그림 11은 보정을 하고자하는 영역을 동심원으로 나타내었고, 표 2는 그림 11의 영역을 구분하는 조건을 보여준다. 그림 9에서 설명한 영역에 식 (22)를 적용하는 것은 하드웨어 복잡도가 증가하기 때문에 표 2를 사용하여 원 영역 내부를 5단계로 나누었고, 타겟 픽셀이 5개의 영역 중에 어떤 영역에 존재하는지 구분하여 각 영역 마다 적절한 보정율을 적용하였다[14].

그림 12와 그림 13은 각각 명도와 색상의 5 영역에 따른 보정율 수치를 나타내었고, 그림에서 실선과 파선은 각각

적목의 정도가 강한 경우와 적목의 정도가 약한 경우의 보정율 수치이다.

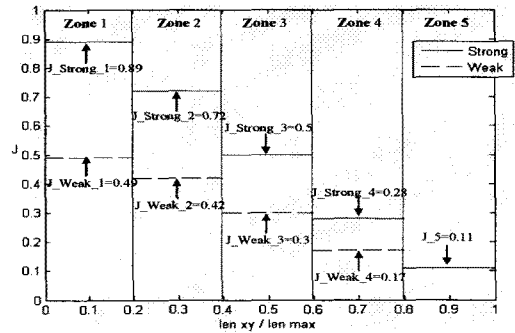


그림 13. 색상 보정을 위한 수치
Fig. 13. Values for color correction

그림 14는 하드웨어 구현을 위한 알고리즘이다. 보정 대상 영역이 피부 영역을 침범할 수 있기 때문에, 그 영역이 피부영역에 해당하는지 확인하여 피부에 해당하는 색이 아닐 경우만 보정을 수행한다.

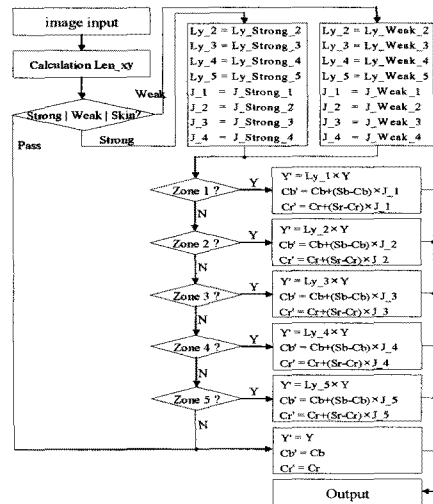


그림 14. 적목 보정을 위한 수치 적용 알고리즘
Fig. 14. Algorithm of applying values

III. 시뮬레이션 결과

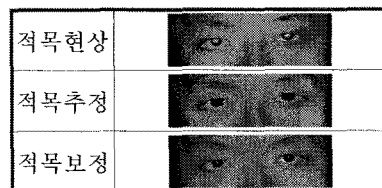


그림 15. 적목현상 보정 시뮬레이션 결과
Fig. 15. The result of redeye correction

본 논문에서 제안하는 알고리즘은 적목현상을 영상처리 기법을 통해서 교정해 내는 것이다. 일반 디지털 카메라(캐논 IXUS 800 IS)로 촬영하여 얻은 적목현상 이미지로 시뮬레이션 하였다. 앞에서 잠깐 언급했듯이, 시뮬레이션은 빛이 없

는 매우 어두운 실내 환경에서 1~1.5m 거리에 떨어진 피사체를 촬영한 이미지를 사용하였다. 제안한 알고리즘은 주로 색을 이용한 마스크를 사용하여 적목현상 영역을 시스템이 자동으로 추정하기 때문에 적목이외에 다른 영역을 적목현상 영역으로 추정하거나, 적목현상 영역을 정확히 추정하지 못하는 경우가 발생할 수 있다. 따라서 시뮬레이션 결과는 적목현상 영역을 정확히 추정하여 보정 결과가 자연스러운 결과만 적목현상을 보정하였다고 판단하였다. 그림 15는 시뮬레이션 예시를 나타내었고, 표 3은 405장의 이미지를 테스트하여 자연스러운 보정 결과만을 모은 데이터이다.

표 3. 시뮬레이션 결과
Table 3. Results of simulation

인원[명]	눈(이미지)[#]	검출[#]	[%]
5 이하	74(20)	50	67.568
9	85(5)	46	54.118
15	246(9)	41	16.667
전체	405(34)	137	33.827

제안한 알고리즘은 하드웨어 설계 시, 하드웨어 자원의 효율성을 고려하여 적목현상 그룹을 최대 32개까지 형성할 수 있도록 하였다. 따라서 이상적인 경우 최대 16명의 적목을 보정할 수 있어야 한다. 하지만 이미지에서 사람의 숫자가 증가할수록 적목의 숫자도 증가할 뿐만 아니라, 적목은 아니지만 적목으로 추정될 수 있는 영역의 숫자 또한 증가하게 된다. 이로 인해, 이미지에 인원이 5명을 초과할 경우 적목보정 성능이 급격히 저하됨이 발견되었으며, 5명 이하의 영상에서 어느 정도의 보정 성능을 보임을 확인하였다. 이러한 성능 감소는 적목현상 추정 영역의 최대 숫자를 증가시킴으로써 일부 해결이 가능하지만 성능을 더욱 향상시키기 위하여 정확하게 적목현상을 추정하는 방법에 대한 연구가 더욱 필요하다.

IV. 하드웨어 설계 결과

설계된 전체 블록의 합성 조건 및 합성 결과는 표 4와 같다. TSMC ASIC Library(0.25um)를 이용하여 Worst Case 조건을 이용하여 합성한 결과 약 31만개의 gate counts로 구현할 수 있었으며 마스크를 생성하여 적목현상을 자동으로 추정하고 보정하는 일련의 과정들이 41.8MHz의 동작 속도를 만족시키는 결과를 얻을 수 있었다. 표 4에서 'Mem Control'블록은 적목보정에 사용되는 2M의 프레임메모리를 제어하고, 시스템의 전반적인 흐름을 조절하는 역할을 한다. 'Pre ARTS'와 'ARTS'블록은 각각 마스크들을 생성하고 그 결과를 이용하여 적목현상으로 예상되는 그룹을 설정하는 역할을 한다. 그리고 'White Check'블록은 사람의 눈 영역에서 흰자위 부분을 검출하는 역할을 한다. 마지막으로 'Concentric Cloud'블록은 제안한 알고리즘의 핵심으로써 앞 블록들의 결과를 이용하여 적목현상을 추정하고 자연스럽게 보정해주는 역할을 수행한다.

표 4. 합성 조건 및 합성 결과 Gate counts
Table 4. Synthetic condition and gate counts

Synopsis Design Analyzer	
Library	TSMC 0.25um ASIC Library
Operation Condition	Worst case
Gate Counts [EA]	
Mem Control	3,814
Pre ARTS	73,745
ARTS	57,137
White Check	34,641
Concentric Cloud	145,814
Total	315,151

V. 결론

본 논문에서는 적목현상 발생 위치 추정과 보정에 대해 언급하였다. 적목의 위치를 추정하기 위해 적목의 색상에 해당하는 픽셀을 추출한 마스크와 피부색상에 해당하는 픽셀을 추출한 마스크 그리고 수정체에 반사되는 광원에 해당하는 픽셀을 추출한 마스크를 생성하였다. 이러한 세 가지 마스크를 이용하여 적목현상이 발생하였을 가능성이 큰 픽셀을 추출하는 방법을 제시하였다. 또한 본 논문에서는 하드웨어 공간의 절약을 위해 근접한 픽셀들은 하나의 그룹으로 생성하도록 그룹화 알고리즘을 제안하였고, 생성된 그룹은 좌우에 흰자위의 색상에 해당하는 픽셀의 존재 여부에 따라 적목을 구분함으로써 적목의 추출 성능을 향상시킬 수 있었다. 자연스러운 보정을 위해 그룹의 가로와 세로의 평균을 지름으로 하는 원을 사용하여 원 내부의 픽셀을 보정하였고, 중심점과 보정 대상 픽셀의 거리와 보정 대상 픽셀의 색상 정도에 따라 다른 보정율을 적용하여 더욱 자연스러운 보정을 할 수 있었다.

본 논문에서 제안한 알고리즘을 405장의 적목현상이 발생한 이미지에 적용하여 simulation한 결과 33.8%정도의 보정 결과를 확인 할 수 있었으며 5명이하의 영상 20장에서는 67.5%의 보정 결과를 얻을 수 있었다. 더욱 자연스러운 보정을 위해서 피부와 적목을 더 정확하게 구분하기 위한 연구가 필요하다.

참고 문헌

- [1] Image Sensor의 시장 동향, 지식경제부, 정보통신연구원, 정보서비스단 통계분석팀, Aug. 2008.
- [2] L. Harrich and R.M. Grant, "The red-eye effect," Physics Teachers, pp. 514-515, Nov. 1985.
- [3] J.H. Alligood and J.S. Lawther, "Compact camera with flash unit movable to reduce red-eye," US Patent 4,855,771, Aug. 1989.
- [4] O. Maida and H. Ogawa, "Flash exposure apparatus for avoiding red-eye effect," US Patent 5,153,632, Jun. 1992.
- [5] S.R. Yhann and T.Y. Young, "Boundary Localization in Texture Segmentation," IEEE Trans. on Image Processing, Vol. 4, No. 6, pp. 849-856, Jun. 1995.

[6] A. Kaufmann and M.M. Gupta, Introduction to Fuzzy Arithmetic Theory and Application, Van Nostrand Reinhold, 1985.

[7] S. Lee, W. Jang, K. Kim, S. Kim and B. Kang, "Real Time Apparatus for Red Eye Correction of 2D Image," ICEIC2008(The 9th Intl. Conf. of Electronics, Information, and Communication), pp. 69-72, Jun. 2008.

[8] K. Jack, Video Demystified, LLH Technology Publishing, pp. 39-62, 2001.

[9] S.S. Wilson, "Theory of Matrix Morphology," IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol. 14, No. 6, pp. 636-652 Jun. 1992.

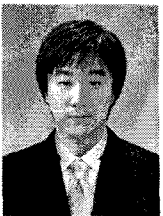
[10] W. Jang, C.S. Lee, S. Kim and B. Kang, "Automatic Red-eye Detection and Correction System for Mobile Devices," Intl. SoC Design Conf.(ISOC 2008) CDC, pp. 5-6, Nov. 2008.

[11] 임정욱, 송진근, 하주영, 강봉순, "소형 DISPLAY 장치를 위한 비 메모리 피부 검출 알고리즘 및 HARDWARE 구현," 한국해양정보통신학회논문지, Vol. 11, No. 8, pp. 1456-1464, Aug. 2007.

[12] R.C. Gonzalez, R.E. Woods, S.L. Eddins, Digital Image Processing Using MATLAB, Prentice-Hall, pp. 199-204, 2004.

[13] 장원우, 이성진, 최원태, 김석찬, 강봉순, "Progressive한 영상을 위한 실시간 그룹화 알고리즘," (사)한국정보기술학회, 2008년도 한국정보기술학회 IT 기반 융합서비스 워크샵 및 하계종합학술대회, pp. I-03, May 2008.

[14] 이성진, 장원우, 김강주, 강봉순, "하드웨어적 적목 현상 교정 시스템," (사)한국정보기술학회, 2008년도 한국정보기술학회 IT 기반 융합서비스 워크샵 및 하계종합학술대회, pp. VIII-01 May 2008.



이 성 진(Chris Songjin Lee)
 2007년 2월 동아대학교 전기전자컴퓨터 공학부 전기공학과(공학사)
 2007년 3월~현재 동아대학교 전자공학과 석사과정

관심분야 : 영상신호처리, SoC설계 및 무선통신



장 원 우(Wonwoo Jang)
 2005년 2월 동아대학교 전기전자컴퓨터 공학부 전자공학과(공학사)
 2007년 2월 동아대학교 전자공학과(공학석사)

2007년 3월~현재 동아대학교 전자공학과 박사과정
 관심분야 : 영상신호처리, SoC설계 및 무선통신



최 원 태(Wontae Choi)
 1986년 8월 경북대학교 전자공학과(공학사)
 1994년 8월 포항공과대학교 전자공학과(공학석사)

1986년~1991년 삼성전자 반도체 선임연구원
 1998년~현재 삼성전기 중앙연구소 수석연구원
 관심분야 : Camera module for mobile phone



김 석 찬(Sukchan Kim)
 1993년 2월 부산대학교 전자공학과(공학사)
 1995년 2월 한국과학기술원 전기및전자공학과(공학석사)

2000년 2월 한국과학기술원 전기및전자공학과(공학박사)
 2000년 한국전자통신연구소(ETRI) 연구원
 2000년~2001년 Princeton 대학교 박사 후 과정
 2001년~2002년 Lehigh 대학교 박사 후 과정
 2002년~2006년 부산대학교 전자공학과 조교수
 2006년~현재 부산대학교 전자공학과 부교수
 2008년~현재 부산대학교 반도체설계교육센터장
 관심분야 : 이동통신, 통신신호처리, OFDM



강 봉 순(Bongsoon Kang)
 1985년 연세대학교 전자공학과(공학사)
 1987년 미국 University of Pennsylvania 전기공학과(공학석사)
 1990년 미국 Drexel University 전기 및 컴퓨터공학과(공학박사)

1989년~1999년 삼성전자 반도체 수석연구원
 1999년~현재 동아대학교 전자공학과 부교수
 2006년~현재 멀티미디어 연구센터 소장
 2006년~현재 2단계 BK21 사업팀장
 관심분야 : 영상신호처리, SoC설계 및 무선통신