

# 이동 전화 카메라를 위한 저조도 환경에서의 영상 번짐 방지 방법

이종호\*, 신인용\*, 이현구\*\*, 김태윤\*\*, 호효성\*

\*광주과학기술원 정보통신공학과, \*\*LG전자 MC 사업부 연구소

## Anti-Blurring Algorithm for Mobile Phone Camera In Dim Lighting Conditions

Jong-Ho Lee\*, In-Yong Shin\*, Hyun-Goo Lee\*\*, Tae-Yoon Kim\*\*, Yo-Sung Ho\*

\*Gwangju Institute of Science and Technology (GIST)

\*\*Mobile Handset R&D Center LG Electronics Inc.

{jongho, siy0808, hoyo}@gist.ac.kr, \*\*{dennylee, jrock}@lge.com

**Abstract:** 저조도 환경에서 이동 전화 카메라로 사진을 촬영하는 경우에, 손 떨림에 의한 번짐 영상을 얻기 쉽다. 현재 상용되고 있는 카메라들에 적용되는 손 떨림 방지 기능의 경우, 하드웨어적 방식이기 때문에 비용과 부피를 고려했을 때 이동전화 카메라에는 적합하지 않다. 흔들린 영상을 보정하는 영상 흐림 보정(image deblurring) 기법의 경우에도, 동작시간과 알고리즘 적용 후의 결과를 보면, 이동전화 카메라에 적용하기에는 무리가 있다. 본 논문에서는 빠른 서터 속도로 찍은 전체 크기 영상(full-size image)과 사진 보기 영상(preview image)을 이용하여, 비용과 부피의 제약을 받지 않으면서도 빠르게 동작하면서, 영상의 번짐을 효과적으로 막는 방법을 제안한다. 1600 x 1200 크기를 갖는 3.66MByte의 영상으로 실험한 결과, 제안한 방법의 총 동작 시간은 1.47초였으며 65.2%의 번짐 보정률을 보였다.

라면, 영상 흐림 보정(image deblurring)의 경우는 이미 번진 영상을 가지고 선명한 영상을 복원하는 방식이다. 번진 영상은 다음과 같이 모델링 할 수 있다.

$$B = P \otimes I + N \quad (1)$$

여기서  $B$ 는 번진 영상,  $I$ 는 점 확산 함수(point spread function),  $P$ 는 원래의 선명한 영상,  $N$ 은 잡음을 나타낸다. 영상 흐림 보정의 경우 번진 영상으로부터 점 확산 함수와 선명한 영상을 구해야 하는 잘못 설정된 문제(ill-posed problem)이기 때문에, 선명한 영상을 구하는 과정에서 많은 반복 연산을 필요로 하고 동작 시간도 길어지게 된다.

Fergus가 제안한 알고리즘의 경우, 128 x 128의 영상에서 약 10분 정도의 시간이 소요되며[1], Shan의 알고리즘의 경우 링잉(ringing) 현상을 효과적으로 줄였지만 동작 시간은 더욱 길어져 1600 x 1200 크기의 영상의 경우 약 20분 정도의 시간이 소요된다[3]. 두 장의 영상을 이용하여 흔들린 영상을 복원하는 알고리즘도 제안되었다[2]. 서터를 강제로 빠르게 해서 얻은, 번짐은 없지만 잡음이 심하고 어두운 영상과 잡음도 없고 올바른 색상 정보를 갖고 있지만 번진 영상. 이렇게 두 장을 가지고 점 확산 함수를 추정하여 이를 이용해 원래의 선명한 영상을 복원하는 방식이다. 이 알고리즘의 경우, 링잉 현상도 많이 개선되었고 번짐 현상도 효과적으로 제거 되었지만, 두 장의 영상을 이용하기 때문에 알고리즘 구현 시 많은 메모리 공간이 필요하다는 단점이 있다. 따라서, 이동 전화 카메라에서 영상의 번짐을 제거하기 위해 영상 흐림 보정 기법을 이용하는 것은 긴 동작 시간과 메모리 문제로 인해서 적합하지 못하다.

**Keywords:** 손 떨림 방지, 이동 전화 카메라

### I. 서론

저조도 환경에서 사진을 촬영할 경우, 충분한 양의 빛을 확보하기 위해서 카메라 셔터를 누른 순간부터 사진이 찍히기까지 오랜 시간이 소요되게 된다. 이 시간이 길어질수록 손 떨림에 의해 번진 영상을 얻을 확률은 높아지게 된다.

시중에 나와있는 많은 카메라들의 경우, 이러한 손 떨림 효과를 방지하기 위한 기능이 내장되어 있다. Canon 사의 영상 안정화 장치(Image stabilizer)의 경우 카메라의 움직임을 포착하여 이를 보상하는 방향으로 렌즈 군을 움직여 영상의 번짐을 막고, Konica Minolta사의 손 떨림 보정 장치(Anti-shake)의 경우 CCD를 움직여 카메라의 흔들림을 보상한다. 하지만 이러한 방식들은, 하드웨어 소자를 움직여 카메라의 움직임을 보상하는 방식이기 때문에 카메라 내부에서 차지하는 부피도 클 뿐만 아니라 구현하는데 비용도 많이 든다. 때문에 비용과 크기에 제약이 따르는 이동 전화의 경우에는 이러한 방식이 적합하지 않다.

하드웨어적인 방식이 카메라의 움직임을 감지하여 이를 보상함으로써, 사진에 영상의 번짐을 막는 방식이

우리가 제안하는 알고리즘은 이러한 문제점들을 해결하여, 비용과 크기의 제약으로부터 자유로우면서도 빠른 동작 속도를 보여준다. 일단, 서터 속도를 빠르게 하여 손 떨림이 최대한 배제된 전체 크기(full-size) 영상을 얻은 뒤, 사진 보기(preview) 영상을 참조하여, 전체 크기 영상에 있는 잡음을 제거하고 색상을 보정하여 원본 영상과 가까운 선명한 영상을 얻을 수 있다.

II. 영상 변질 방지 알고리즘

그림 1에 전체적인 영상 변질 방지 알고리즘을 나타내었다.

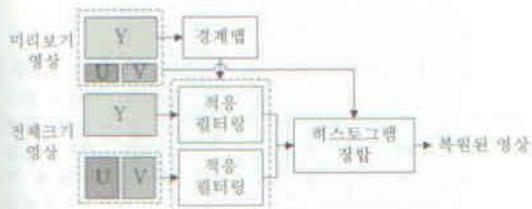


그림 1. 제안하는 알고리즘

우선, 셔터 속도를 빠르게 해서 전체 크기 영상을 얻은 뒤, 사진 보기 영상을 이용하여 전체 크기 영상의 잡음을 제거한다. 잡음이 제거 된 전체 크기 영상과 사진 보기 영상을 이용하여 히스토그램 정합을 통해 전체 크기 영상의 색상을 보정하여 번짐이 없는 선명한 영상을 얻는다.

1. 입력 영상의 획득

우선, 손 필립에 의한 영상의 번짐을 사전에 최대한 방지하기 위해 셔터 속도를 빠르게 해서 전체 크기 영상을 얻는다. 이 때의 셔터 속도는 70ms이며, 전체 크기 영상의 경우 1600x1200 크기의 YUV 4:2:2 포맷을 갖는다. 전체 크기 영상의 경우, 번짐 현상은 거의 없지만, 영상이 어둡고 잡음이 많이 섞여있다. 카메라 내부에서 사진 보기 영상도 이용 가능한데, 사진 보기 영상의 경우 320x240 크기의 YUV 4:2:0의 포맷을 갖는다. 사진 보기 영상의 경우 크기는 작지만 잡음이 없고 올바른 색상 정보를 갖고 있다.

2. 적응 양방향 필터링

셔터 속도를 빠르게 해서 사진을 촬영할 경우, 빛의 부족으로 인해 신호의 세기가 약해지고, 카메라 내부에 존재하는 잡음 flow가 상대적으로 두드러져, 잡음이 섞인 영상이 촬영되게 된다.

이러한 잡음을 제거하기 위하여 양방향 필터를 기본 필터로 사용하였다. 최근에 양방향 필터보다 잡음 제거 성능이 우수한 Non local means 알고리즘이 소개되었으나[5] 동작 시간이 너무 긴 단점이 있어 비교적 동작 속도도 빠르면서 경계부분을 잘 살리는 동시에 잡음제거 효과도 뛰어난 양방향 필터를 이용하였다.

$$F(I)_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_p}(\|p - q\|) G_{\sigma_s}(\|I_p - I_q\|) I_q \quad (2)$$

여기서  $I$ 는 필터의 입력 영상,  $F(I)$ 는 필터의 결과 영상,  $p$ 는 복원하고자 하는 화소,  $S$ 는  $p$ 를 중심으로 하는 윈도우,  $q$ 는  $S$ 에 속하는  $p$ 의 이웃 화소,  $I_p$ 는  $p$ 화소의 화소 값,  $I_q$ 는  $q$ 화소의 화소 값  $G$ 는 가우시안 함수,  $\sigma_p$ 는 화소 사이 거리를 인자로 하는 가우시안 함수의 표준편차,  $\sigma_s$ 는 화소 사이 크기 차를 인자로 하는 가우시안 함수의 표준편차,  $W_p$ 는 정규화 상수를 나타낸다. 즉, 잡

음이 섞인 영상에서 각각의 화소의 복원은, 복원하고자 하는 화소를 중심으로 하는 윈도우 내에 있는 이웃 화소들의 가중 합으로 구해진다. 이때, 각 이웃 화소들의 가중치는 복원하고자 하는 화소와 이웃 화소 사이의 크기 차이와 거리를 인자로 하는 두 개의 가우시안 함수를 곱하여 얻어진다. 양방향 필터링의 경우, 윈도우 크기,  $\sigma_p$  그리고  $\sigma_s$  이렇게 3개의 매개변수를 통하여 필터링이 수행되는데, 영상의 지역적 특성을 고려하여 영역에 따라 서로 다른 매개변수를 가지고 필터링을 수행하면, 경계부분은 잘 살리면서 잡음을 더욱 효과적으로 제거할 수 있다.

이렇게 영상의 지역적 특성에 따라 다른 매개변수로 적응 필터링을 하기 위해선 영상의 경계정보가 필요하다. 하지만 경계와 잡음 모두 높은 주파수 성분을 갖기 때문에, 잡음이 많이 섞여 있는 영상에서 경계정보를 효과적으로 추출하기가 쉽지 않다. 그래서 우리는 잡음이 없는 사진보기 영상을 이용하여 경계정보를 추출하였다. 사진보기 영상을 이용하여 경계정보를 추출하기가 작기 때문에 간단한 Sobel 알고리즘을 통해서도 비교적 빠르고 정확하게 경계 정보를 검출할 수 있다.

적용 필터링을 위해, 사진보기 영상의 Y성분으로부터 추출한 경계맵을 이용하여 필터링 영역을 3개의 영역으로 구분한다. 경계맵 상에서, 3x3윈도우를 움직여 가며 경계가 없는 영역(영역 1), 강한 경계가 있는 영역(영역 2), 많은 경계가 포함된 영역(영역 3)을 구분한 뒤, 구분한 영역에 따라 각기 다른 매개변수를 이용하여 필터링을 한다. 사진 보기 영상과 전체 크기 영상의 크기 차이로 인해, 경계맵 상에서의 1 화소는 전체 크기 영상에서는 5x5 블록에 해당한다. 휘도 신호보다는 색차 신호에 포함 된 잡음의 경우 인간의 눈에 더 잘 띄기 때문에, Y성분과 U, V성분을 나누어 따로 필터링한다. 영역에 따른 Y성분과 U, V성분의 필터링 매개변수를 표 1에 나타내었다.

표 1. 영역 구분에 따른 필터링 매개변수

성분	영역구분	윈도우 크기	$\sigma_p$	$\sigma_s$
Y	영역 1	3x3	1	50
	영역 2	3x3	1	50
	영역 3	3x3	1	50
U,V	영역 1	11x11	1	50
	영역 2	7x7	1	30
	영역 3	3x3	1	5

3. 히스토그램 정합

영상의 밝기를 보정하기 위해 히스토그램 정합을 이용하였다. 사진보기 영상의 경우 올바른 밝기 정보를 갖고 있기 때문에 잡음 제거 시와 마찬가지로 히스토그램 정합 시에도 사진 보기 영상을 참조하여 밝기를 보정한다.

전체 크기 영상을 사진보기 영상의 크기로 다운샘플링한 후, 전체 크기 영상과 사진 보기 영상으로부터 히스토그램을 추출한다. 추출한 두 개의 히스토그램으로부터 그림 2(a)와 같이 두 개의 누적 히스토그램을 구한 뒤, 이로부터 매핑 함수를 구하여 전체 크기 영상의 화소값들을 밝은 값들로 대체한다. 하지만 이렇게 단순히 매핑 함수를 이용하여 화소값들을 보정할 경우,

그림 2(b)에 보이듯이 셔터 속도를 빠르게 해서 어두운 영상이 찍힐 때, 0으로 포화되었던 모든 화소 값들이 일제히 동일한 값으로 보정되기 때문에 원래의 영상과 다른 부자연스러운 영상을 얻게 된다. 이 문제를 해결하기 위해서 우리는 Fecker의 방법을 이용하였다 [4].

$$c = \frac{\sum_{i=0}^{M[0]} iH[i]}{\sum_{i=0}^{M[0]} H[i]} \quad (3)$$

$M[0]$  대신에 이 값을 0의 매핑 값으로 대체한다. 그림 2(c)에 무게중심을 이용하여 새롭게 매핑한 결과 영상의 히스토그램을 나타내었다. 하지만 이렇게 무게중심을 이용하여 0의 매핑 값을 보정을 하더라도, 그림 2(c)에 보듯이 첫 번째 히스토그램 막대와 두 번째 히스토그램 막대 사이에 부자연스러운 빈 간격이 발생하게 된다. 그래서 우리는 히스토그램을 촘촘히 재배열하여 영상의 화질을 더욱 개선하였다. 그림 2(c)의 히스토그램에서 총  $2(M[0]-c)$ 개의 히스토그램 막대를 다음의 식을 이용하여 재배열하였다.

$$M'[i] = M[i] - \frac{(M[0]-c)(2(M[0]-c)-i)}{2(M[0]-c)} \quad (4)$$

$$i = 0 \dots 2(M[0]-c) - 1$$

여기서  $M'[i]$ 는 재배열된 매핑 결과,  $M[i]$ 는 전체 크기 영상의 매핑값,  $c$ 는 무게중심을 나타낸다. 재배열된 히스토그램을 그림 2(d)에 나타내었다.

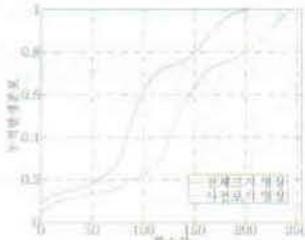
### III. 실험 결과

알고리즘의 효과를 비교하기 위해, 정지된 영상을 41Hz의 속도로 수직 방향으로 움직이는 장치에 카메라를 부착하고 실험하였다. 영상 번짐 방지기능을 활성화시킨 상태에서 얻은 전체 크기 영상과, 사전 보기 영상을 이용해 알고리즘을 수행한 뒤 번짐 방지기능을 활성화시키지 않은 상태에서 얻은 영상과 비교해 보았다.

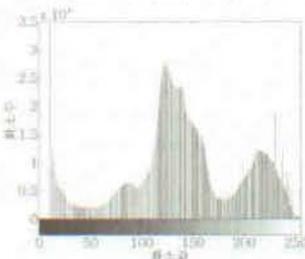
알고리즘의 실행은 3GB의 메모리를 갖는 펜티엄 4 3.4GHz 프로세서로 실험하였으며, 알고리즘의 총 동작 시간은 1.47초였다. 입력 영상으로는 70ms의 빠른 셔터 속도로 찍힌 1600x1200 크기의 전체 크기 영상과, 320x240 크기의 사전 보기 영상이 이용되었다. 전체 크기 영상으로부터 잡음을 제거한 뒤, 히스토그램 정합을 통해 색상을 보정한 결과들을 영상의 일부를 확대해서 그림 3에 나타내었다.

번짐 방지 기능을 활성화하지 않고 움직이는 카메라로 촬영했을 경우와, 번짐 방지 기능을 활성화하고 나서 움직이는 카메라로 촬영했을 경우 각각의 영상을 그림 4에 나타내었다.

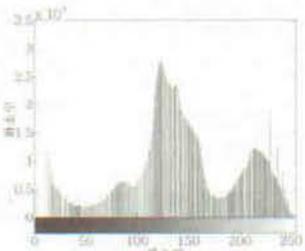
본 논문에서 제안한 알고리즘의 성능을 평가하기 위해 시중에 나와 있는 카메라들의 촬영 영상들과 제안한 알고리즘으로 얻은 결과 영상으로부터 번짐 보정률을 측정하여 비교하였다. 번짐 보정률은 번짐 방지 기능을 활성화하기 전과 활성화한 후에 얻은 영상들로부터 각각 번짐 길이를 측정하고 이로부터 식 (5)를 이용하여 보정률을 계산하였다. 번짐 길이의 측정을 위하여 우선, 영상의 경계 부분에서 경계부분을 수직으로 가로지르는 방향으로 휘도 곡선을 추출하여 휘도 곡선의 최소값과 최대값을 계산한다. 휘도 곡선 상에서 최소값



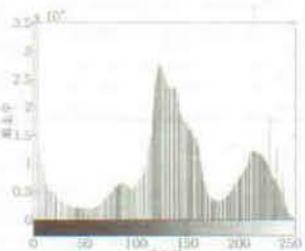
(a) 사전보기 영상과 전체 크기 영상의 누적 히스토그램



(b) 매핑함수에 의해 단순 매핑한 결과



(c) 무게 중심을 구하여 보정한 결과



(d) 히스토그램 재배열을 통해 보정한 결과

### 그림 2. 색상 보정 과정에서의 히스토그램

단순 매핑 함수에 의해 화소값 0이 매핑되는 값을  $M[0]$ , 사전 보기 영상에서 화소값  $i$ 의 히스토그램을  $H[i]$ 라 하면, 무게중심  $c$ 를 다음과 같이 구할 수 있다.

+0.1x 최대값과 최대값-0.1x 최대값을 갖는 화소의 위치를 각각 구하여 이 두 화소 사이의 거리를 변진 길이로 계산한다.

$$\text{변진 보정률(\%)} = (1 - \frac{l_B'}{l_B}) \times 100 \quad (5)$$

여기서  $l_B'$  와  $l_B$ 는 변진 방지 기능을 활성화한 경우와 활성화하지 않았을 때 변진 길이를 각각 나타낸다. 표 2에 시중에 나와있는 카메라들과 제안한 알고리즘을 이용했을 때 각각의 변진 보정률을 나타내었다.

표 2. 상용 카메라들과 변진 보정률 비교

	카메라 A	카메라 B	카메라 C	제안방법
변진 길이 (cm)	0.186	0.127	0.154	0.074
변진 보정률 (%)	8.6	42.5	45.5	65.2

\* 정확한 모델명은 익명성을 위해 알파벳으로 대체함

#### IV. 결론

본 논문에서는 저조도 환경에서 이동 전화 카메라로 촬영을 할 경우 생기는 영상의 변진을 방지하는 알고리즘을 제안하였다. 셔터 속도를 빠르게 해서 얻은 전체 크기 영상으로부터, 미리 보기 영상을 참조하여 잡음을 제거하고, 색상을 보정하여 선명한 영상을 얻는다. 제안한 알고리즘의 경우 비용과 부피의 제약을 받지 않으면서도 총 동작 시간이 1.47초로 수행 속도가 빠르고 65.2%의 변진 보정률로 영상의 변진도 효과적으로 줄이는 것을 확인할 수 있다.

#### 감사의 글

\*본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과(GITA-2009-(C1090-0902-0017))와 LG 전자 MC 사업부 연구소의 지원으로 수행되었음\*

#### 참고문헌

- [1] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis and W. Freeman, "Removing camera shake from a single photograph," *ACM Transactions on Graphics*, vol. 25, pp.787-794, July 2006
- [2] L. Yuan, J. Sun, L. Quan and H.-Y. Shun, "Image deblurring with blurred/noisy image pairs," *ACM SIGGRAPH*, vol. 26, pp.1-10, May 2007
- [3] Q. Shan, J. Jia and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Transactions on Graphics*, vol. 27, pp.1-10, August 2008
- [4] U. Fecker, M. Barkowsky and A. Kaup, "Improving the prediction efficiency for multi-view video coding using histogram matching," *In Proceedings of the Picture Coding Symposium*, pp.2-16, April 2006
- [5] A. Buades, B. Coll, J.-M. Morel, "A non-local algorithm for image denoising," *Computer Vision and Pattern Recognition*, vol. 2, pp.60-65, June 2005



(a) 셔터 속도를 빠르게 해서 얻은 영상

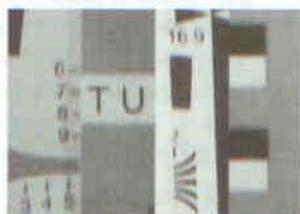


(b) 잡음을 제거한 영상



(c) 색상을 보정한 영상

그림 3. 알고리즘 각 단계에서의 결과 영상



(a) 변진 방지 기능을 활성화하지 않았을 경우



(b) 변진 방지 기능을 활성화하였을 경우

그림 4. 변진 방지 기능의 활성화 여부에 따른 영상 비교

# 이동 전화 카메라를 위한 저조도 환경에서의 영상 번짐 방지 방법

이종호\*, 신인용\*, 이현구\*\*, 김태윤\*\*, 호요성\*

\*광주과학기술원 정보통신공학과, \*\*LG전자 MC 사업부 연구소

## Anti-Blurring Algorithm for Mobile Phone Camera In Dim Lighting Conditions

Jong-Ho Lee\*, In-Yong Shin\*, Hyun-Goo Lee\*\*, Tae-Yoon Kim\*\*, Yo-Sung Ho\*

\*Gwangju Institute of Science and Technology (GIST)

\*\*Mobile Handset R&D Center LG Electronics Inc.

\*{jongho, siy0808, hoyo}@gist.ac.kr, \*\*{dennylee, jrock}@lge.com

**Abstract:** 저조도 환경에서 이동 전화 카메라로 사진을 촬영하는 경우에, 손 떨림에 의한 번짐 영상을 얻기 쉽다. 현재 상용되고 있는 카메라들에 적용되는 손 떨림 방지 기능의 경우, 하드웨어적 방식이기 때문에 비용과 부피를 고려했을 때 이동전화 카메라에는 적합하지 않다. 흔들린 영상을 보정하는 영상 흐림 보정(image deblurring) 기법의 경우에도, 동작시간과 알고리즘 적용 후의 결과를 보면, 이동전화 카메라에 적용하기에는 무리가 있다. 본 논문에서는 빠른 셔터 속도로 찍은 전체크기 영상(full-size image)과 사전 보기 영상(preview image)을 이용하여, 비용과 부피의 제약을 받지 않으면서도 빠르게 동작하면서, 영상의 번짐을 효과적으로 막는 방법을 제안한다. 1600 x 1200 크기를 갖는 3.66MByte의 영상으로 실험한 결과, 제안한 방법의 총 동작 시간은 1.47초였으며 65.2%의 번짐 보정률을 보였다.

**Keywords:** 손 떨림 방지, 이동 전화 카메라

### I. 서론

저조도 환경에서 사진을 촬영할 경우, 충분한 양의 빛을 확보하기 위해서 카메라 셔터를 누른 순간부터 사진이 찍히기까지 오랜 시간이 소요되게 된다. 이 시간이 길어질수록 손 떨림에 의해 번진 영상을 얻을 확률은 높아지게 된다.

시중에 나와있는 많은 카메라들의 경우, 이러한 손 떨림 효과를 방지하기 위한 기능이 내장되어 있다. Canon 사의 영상 안정화 장치(Image stabilizer)의 경우 카메라의 움직임을 포착하여 이를 보상하는 방향으로 렌즈 군을 움직여 영상의 번짐을 막고, Konica Minolta사의 손 떨림 보정 장치(Anti-shake)의 경우 CCD를 움직여 카메라의 흔들림을 보상한다. 하지만 이러한 방식들은, 하드웨어 소자를 움직여 카메라의 움직임을 보상하는 방식이기 때문에 카메라 내부에서 차지하는 부피도 클 뿐만 아니라 구현하는데 비용도 많이 든다. 때문에 비용과 크기에 제약이 따르는 이동 전화의 경우에는 이러한 방식이 적합하지 않다.

하드웨어적인 방식이 카메라의 움직임을 감지하여 이를 보상함으로써, 사전에 영상의 번짐을 막는 방식이

라면, 영상 흐림 보정(image deblurring)의 경우는 이미 번진 영상을 가지고 선명한 영상을 복원하는 방식이다. 번진 영상은 다음과 같이 모델링 할 수 있다.

$$B = P \otimes I + N \quad (1)$$

여기서  $B$ 는 번진 영상,  $P$ 는 점 확산 함수(point spread function),  $I$ 는 원래의 선명한 영상,  $N$ 은 잡음을 나타낸다. 영상 흐림 보정의 경우 번진 영상으로부터 점 확산 함수와 선명한 영상을 구해야 하는 잘못된 설정된 문제(ill-posed problem)이기 때문에, 선명한 영상을 구하는 과정에서 많은 반복 연산을 필요로 하고 동작 시간도 길어지게 된다.

Fergus가 제안한 알고리즘의 경우, 128 x 128의 영상에서 약 10분 정도의 시간이 소요되며[1], Shan의 알고리즘의 경우 링잉(ringing) 현상을 효과적으로 줄였지만 동작 시간은 더욱 길어져 1600 x 1200 크기의 영상의 경우 약 20분 정도의 시간이 소요된다[3]. 두 장의 영상을 이용하여 흔들린 영상을 복원하는 알고리즘도 제안되었다[2]. 셔터를 강제로 빠르게 해서 얻은, 번짐은 없지만 잡음이 심하고 어두운 영상과 잡음도 없고 올바른 색상 정보를 갖고 있지만 번진 영상, 이렇게 두 장을 가지고 점 확산 함수를 추정하여 이를 이용해 원래의 선명한 영상을 복원하는 방식이다. 이 알고리즘의 경우, 링잉 현상도 많이 개선되었고 번짐 현상도 효과적으로 제거 되었지만, 두 장의 영상을 이용하기 때문에 알고리즘 구현 시 많은 메모리 공간이 필요하다는 단점이 있다. 따라서, 이동 전화 카메라에서 영상의 번짐을 제거하기 위해 영상 흐림 보정 기법을 이용하는 것은 긴 동작 시간과 메모리 문제로 인해서 적합하지 못하다.

우리가 제안하는 알고리즘은 이러한 문제점들을 해결하여, 비용과 크기의 제약으로부터 자유로우면서도 빠른 동작 속도를 보여준다. 일단, 셔터 속도를 빠르게 하여 손 떨림이 최대한 배제된 전체 크기(full-size) 영상을 얻은 뒤, 사전 보기(preview) 영상을 참조하여, 전체 크기 영상에 있는 잡음을 제거하고 색상을 보정하여 원본 영상과 가까운 선명한 영상을 얻을 수 있다.

## II. 영상 변질 방지 알고리즘

그림 1에 전체적인 영상 변질 방지 알고리즘을 나타내었다.

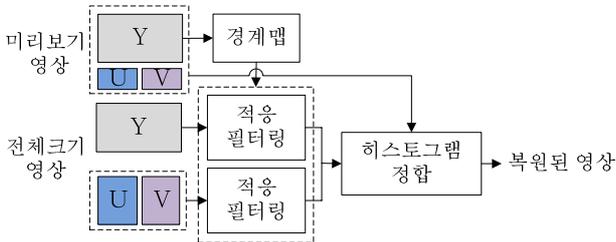


그림 1. 제안하는 알고리즘

우선, 셔터 속도를 빠르게 해서 전체 크기 영상을 얻은 뒤, 사전 보기 영상을 이용하여 전체 크기 영상의 잡음을 제거한다. 잡음이 제거 된 전체 크기 영상과 사전 보기 영상을 이용하여 히스토그램 정합을 통해 전체 크기 영상의 색상을 보정하여 번짐이 없는 선명한 영상을 얻는다.

### 1. 입력 영상의 획득

우선, 손 떨림에 의한 영상의 번짐을 사전에 최대한 방지하기 위해 셔터 속도를 빠르게 해서 전체 크기 영상을 얻는다. 이 때의 셔터 속도는 70ms이며, 전체 크기 영상의 경우 1600 x 1200 크기의 YUV 4:2:2 포맷을 갖는다. 전체 크기 영상의 경우, 번짐 현상은 거의 없지만, 영상이 어둡고 잡음이 많이 섞여있다. 카메라 내부에서 사전 보기 영상도 이용 가능한데, 사전 보기 영상의 경우 320 x 240 크기의 YUV 4:2:0의 포맷을 갖는다. 사전 보기 영상의 경우 크기는 작지만 잡음이 없고 올바른 색상 정보를 갖고 있다.

### 2. 적응 양방향 필터링

셔터 속도를 빠르게 해서 사진을 촬영할 경우, 빛의 부족으로 인해 신호의 세기가 약해지고, 카메라 내부에 존재하는 잡음 flow가 상대적으로 두드러져, 잡음이 섞인 영상이 촬영되게 된다.

이러한 잡음을 제거하기 위하여 양방향 필터를 기본 필터로 사용하였다. 최근에 양방향 필터보다 잡음 제거 성능이 우수한 Non local means 알고리즘이 소개되었으나[5] 동작 시간이 너무 긴 단점이 있어 비교적 동작 속도도 빠르면서 경계부분을 잘 살리는 동시에 잡음제거 효과도 뛰어난 양방향 필터를 이용하였다.

$$F[I]_p = \frac{1}{W_p} \sum_{q \in S} G_{\sigma_s}(\|p - q\|) G_{\sigma_r}(\|I_p - I_q\|) I_q \quad (2)$$

여기서  $I$ 는 필터의 입력 영상,  $F[I]$ 는 필터의 결과 영상,  $p$ 는 복원하고자 하는 화소,  $S$ 는  $p$ 를 중심으로 하는 윈도우,  $q$ 는  $S$ 에 속하는  $p$ 의 이웃 화소,  $I_p$ 는  $p$ 화소의 화소 값,  $I_q$ 는  $q$ 화소의 화소 값  $G$ 는 가우시안 함수,  $\sigma_s$ 는 화소 사이 거리를 인자로 하는 가우시안 함수의 표준편차,  $\sigma_r$ 는 화소 사이 크기 차를 인자로 하는 가우시안 함수의 표준편차,  $W_p$ 는 정규화 상수를 나타낸다. 즉, 잡

음이 섞인 영상에서 각각의 화소의 복원은, 복원하고자 하는 화소를 중심으로 하는 윈도우 내에 있는 이웃 화소들의 가중 합으로 구해진다. 이때, 각 이웃 화소들의 가중치는 복원하고자 하는 화소와 이웃 화소 사이의 크기 차이와 거리를 인자로 하는 두 개의 가우시안 함수를 곱하여 얻어진다. 양방향 필터링의 경우, 윈도우 크기,  $\sigma_s$  그리고  $\sigma_r$  이렇게 3개의 매개변수를 통하여 필터링이 수행되는데, 영상의 지역적 특성을 고려하여 영역에 따라 서로 다른 매개변수를 가지고 필터링을 수행하면, 경계부분은 잘 살리면서 잡음을 더욱 효과적으로 제거할 수 있다.

이렇게 영상의 지역적 특성에 따라 다른 매개변수로 적응 필터링을 하기 위해선 영상의 경계정보가 필요하다. 하지만 경계와 잡음 모두 높은 주파수 성분을 갖기 때문에, 잡음이 많이 섞여 있는 영상에서 경계정보를 효과적으로 추출하기가 쉽지 않다. 그래서 우리는 잡음이 없는 사전보기 영상을 이용하여 경계정보를 추출하였다. 사전보기 영상을 이용할 경우, 잡음이 없고 크기가 작기 때문에 간단한 Sobel 알고리즘을 통해서도 비교적 빠르고 정확하게 경계 정보를 검출할 수 있다.

적응 필터링을 위해, 사전보기 영상의 Y성분으로부터 추출한 경계맵을 이용하여 필터링 영역을 3개의 영역으로 구분한다. 경계맵 상에서, 3x3윈도우를 움직여 가며 경계가 없는 영역(영역 1), 강한 경계가 있는 영역(영역 2), 많은 경계가 포함된 영역(영역 3)을 구분한 뒤, 구분한 영역에 따라 각기 다른 매개변수를 이용하여 필터링을 한다. 사전 보기 영상과 전체 크기 영상의 크기 차이로 인해, 경계맵 상에서의 1 화소는 전체 크기 영상에서는 5x5 블록에 해당한다. 휘도 신호보다는 색차 신호에 포함 된 잡음의 경우 인간의 눈에 더 잘 띄기 때문에, Y성분과 U, V성분을 나누어 따로 필터링한다. 영역에 따른 Y성분과 U, V성분의 필터링 매개변수를 표 1에 나타내었다.

표 1. 영역 구분에 따른 필터링 매개변수

성분	영역구분	윈도우 크기	$\sigma_s$	$\sigma_r$
Y	영역 1	3x3	1	50
	영역 2	3x3	1	50
	영역 3	3x3	1	50
U, V	영역 1	11x11	1	50
	영역 2	7x7	1	30
	영역 3	3x3	1	5

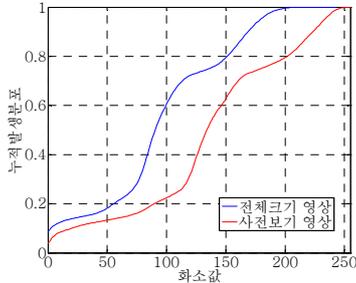
### 3. 히스토그램 정합

영상의 밝기를 보정하기 위해 히스토그램 정합을 이용하였다. 사전보기 영상의 경우 올바른 밝기 정보를 갖고 있기 때문에 잡음 제거 시와 마찬가지로 히스토그램 정합 시에도 사전 보기 영상을 참조하여 밝기를 보정한다.

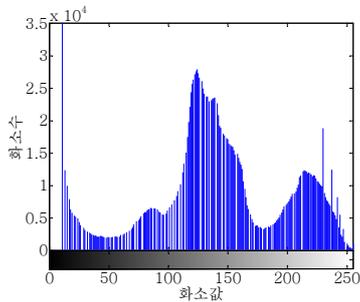
전체 크기 영상을 사전보기 영상의 크기로 다운샘플링한 후, 전체 크기 영상과 사전 보기 영상으로부터 히스토그램을 추출한다. 추출한 두 개의 히스토그램으로부터 그림 2(a)와 같이 두 개의 누적 히스토그램을 구한 뒤, 이로부터 매핑 함수를 구하여 전체 크기 영상의 화소값들을 밝은 값들로 대체한다. 하지만 이렇게 단순히 매핑 함수를 이용하여 화소값들을 보정할 경우,

그림 2(b)에 보이듯이 셔터 속도를 빠르게 해서 어두운 영상이 찍힐 때, 0으로 포화되었던 모든 화소 값들이 일제히 동일한 값으로 보정되기 때문에 원래의 영상과 다른 부자연스러운 영상을 얻게 된다. 이 문제를 해결하기 위해서 우리는 Fecker의 방법을 이용하였다 [4].

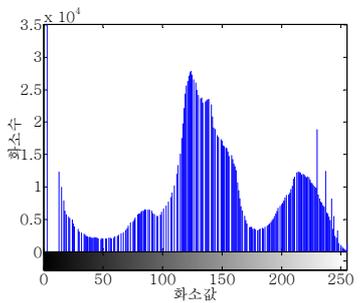
$$c = \frac{\sum_{i=0}^{M[0]} iH[i]}{\sum_{i=0}^{M[0]} H[i]} \quad (3)$$



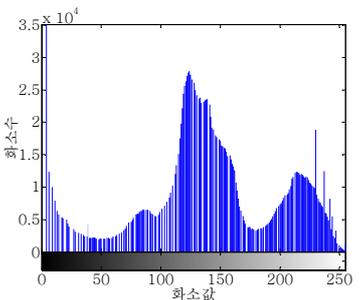
(a) 사전보기 영상과 전체크기 영상의 누적 히스토그램



(b) 매핑함수에 의해 단순 매핑한 결과



(c) 무게 중심을 구하여 보정한 결과



(d) 히스토그램 재배열을 통해 보정한 결과

### 그림 2. 색상 보정 과정에서의 히스토그램

단순 매핑 함수에 의해 화소값 0이 매핑되는 값을  $M[0]$ , 사전 보기 영상에서 화소값  $i$ 의 히스토그램을  $H[i]$ 라 하면, 무게중심  $c$ 를 다음과 같이 구할 수 있다.

$M[0]$  대신에 이 값을 0의 매핑 값으로 대체한다. 그림 2(c)에 무게중심을 이용하여 새롭게 매핑한 결과 영상의 히스토그램을 나타내었다. 하지만 이렇게 무게중심을 이용하여 0의 매핑 값을 보정을 하더라도, 그림 2(c)에 보듯이 첫 번째 히스토그램 막대와 두 번째 히스토그램 막대 사이에 부자연스러운 빈 간격이 발생하게 된다. 그래서 우리는 히스토그램을 촘촘히 재배열하여 영상의 화질을 더욱 개선하였다. 그림 2(c)의 히스토그램에서 총  $2(M[0]-c)$ 개의 히스토그램 막대를 다음의 식을 이용하여 재배열하였다.

$$M'[i] = M[i] - \frac{(M[0]-c)(2(M[0]-c)-i)}{2(M[0]-c)} \quad (4)$$

$$i = 0 \dots 2(M[0]-c) - 1$$

여기서  $M'[i]$ 는 재배열된 매핑 결과,  $M[i]$ 는 전체 크기 영상의 매핑값,  $c$ 는 무게중심을 나타낸다. 재배열된 히스토그램을 그림 2(d)에 나타내었다.

### III. 실험 결과

알고리즘의 효과를 비교하기 위해, 정지된 영상을 4Hz의 속도로 수직 방향으로 움직이는 장치에 카메라를 부착하고 실험하였다. 영상 번짐 방지기능을 활성화시킨 상태에서 얻은 전체 크기 영상과, 사전 보기 영상을 이용해 알고리즘을 수행한 뒤 번짐 방지기능을 활성화시키지 않은 상태에서 얻은 영상과 비교해 보았다.

알고리즘의 실행은 3GB의 메모리를 갖는 펜티엄 4 3.4GHz 프로세서로 실험하였으며, 알고리즘의 총 동작 시간은 1.47초였다. 입력 영상으로는 70ms의 빠른 셔터 속도로 찍힌 1600x1200 크기의 전체 크기 영상과, 320x240 크기의 사전 보기 영상이 이용되었다. 전체 크기 영상으로부터 잡음을 제거한 뒤, 히스토그램 정합을 통해 색상을 보정한 결과들을 영상의 일부를 확대해서 그림 3에 나타내었다.

번짐 방지 기능을 활성화하지 않고 움직이는 카메라로 촬영했을 경우와, 번짐 방지 기능을 활성화하고 나서 움직이는 카메라로 촬영했을 경우 각각의 영상을 그림 4에 나타내었다.

본 논문에서 제안한 알고리즘의 성능을 평가하기 위해 시중에 나와 있는 카메라들의 촬영 영상들과 제안한 알고리즘으로 얻은 결과 영상으로부터 번짐 보정률을 측정하여 비교하였다. 번짐 보정률은 번짐 방지 기능을 활성화하기 전과 활성화한 후에 얻은 영상들로부터 각각 번진 길이를 측정하고 이로부터 식 (5)를 이용하여 보정률을 계산하였다. 번진 길이의 측정을 위하여 우선, 영상의 경계 부분에서 경계부분을 수직으로 가로지르는 방향으로 휘도 곡선을 추출하여 휘도 곡선의 최소값과 최대값을 계산한다. 휘도 곡선 상에서 최소값

+0.1 x 최대값과 최대값-0.1 x 최대값을 갖는 화소의 위치를 각각 구하여 이 두 화소 사이의 거리를 번짐 길이로 계산한다.

$$\text{번짐 보정률(\%)} = (1 - \frac{l_B'}{l_B}) \times 100 \quad (5)$$

여기서  $l_B'$  와  $l_B$ 는 번짐 방지 기능을 활성화한 경우와 활성화하지 않았을 때 번짐 길이를 각각 나타낸다. 표 2에 시중에 나와있는 카메라들과 제안한 알고리즘을 이용했을 때 각각의 번짐 보정률을 나타내었다.

표 2. 상용 카메라들과 번짐 보정률 비교

	카메라 A	카메라 B	카메라 C	제안방법
번짐 길이 (cm)	0.186	0.127	0.154	0.074
번짐 보정률 (%)	8.6	42.5	45.5	65.2

\* 정확한 모델명은 익명성을 위해 알파벳으로 대체함

#### IV. 결론

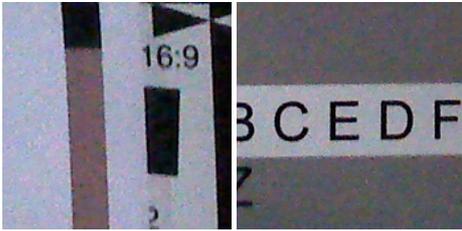
본 논문에서는 저조도 환경에서 이동 전화 카메라로 촬영을 할 경우 생기는 영상의 번짐을 방지하는 알고리즘을 제안하였다. 셔터 속도를 빠르게 해서 얻은 전체 크기 영상으로부터, 미리 보기 영상을 참조하여 잡음을 제거하고, 색상을 보정하여 선명한 영상을 얻는다. 제안한 알고리즘의 경우 비용과 부피의 제약은 받지 않으면서도 총 동작 시간이 1.47초로 수행 속도가 빠르고 65.2%의 번짐 보정률로 영상의 번짐도 효과적으로 줄이는 것을 확인할 수 있다.

#### 감사의 글

"본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과(IITA-2009-(C1 090-0902-0017))와 LG 전자 MC 사업부 연구소의 지원으로 수행되었음"

#### 참고문헌

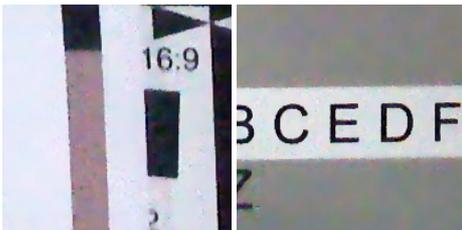
- [1] R. Fergus, B. Singh, A. Hertzmann, S. T. Roweis and W. Freeman, "Removing camera shake from a single photograph," *ACM Transactions on Graphics*, vol. 25, pp.787-794, July 2006
- [2] L. Yuan, J. Sun, L. Quan and H.-Y. Shun, "Image deblurring with blurred/noisy image pairs," *ACM SIGGRAPH*, vol. 26, pp.1-10, May 2007
- [3] Q. Shan, J. Jia and A. Agarwala, "High-quality motion deblurring from a single image," *ACM Transactions on Graphics*, vol. 27, pp.1-10, August 2008
- [4] U. Fecker, M. Barkowsky and A. Kaup, "Improving the prediction efficiency for multi-view video coding using histogram matching," *In Proceedings of the Picture Coding Symposium*, pp.2-16, April 2006
- [5] A. Buades, B. Coll, J.-M. Morel, "A non-local algorithm for image denoising," *Computer Vision and Pattern Recognition*, vol. 2, pp.60-65, June 2005



(a) 셔터 속도를 빠르게 해서 얻은 영상

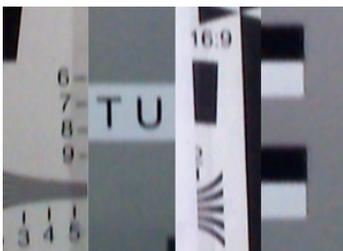


(b) 잡음을 제거한 영상

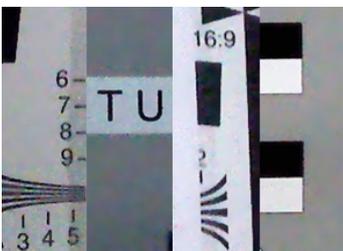


(c) 색상을 보정한 영상

그림 3. 알고리즘 각 단계에서의 결과 영상



(a) 번짐 방지 기능을 활성화하지 않았을 경우



(b) 번짐 방지 기능을 활성화하였을 경우

그림 4. 번짐 방지 기능의 활성화 여부에 따른 영상 비교