

실험을 통한 효과적인 CAPTCHA 시스템을 위한 서브 이미지 추출 개선방안

정우근

Chung Woo-Keun

부산대학교 컴퓨터공학과

wkchung@pusan.ac.kr

ABSTRACT

CAPTCAH (Completely Automated Public Turing test to tell Computers and Humans Apart) 은 스팸이나 로봇에 의한 사이트 자동가입, 계정 생성방지 도구로써, 인간의 우수한 가독성을 통해 특정 언어 또는 그림을 해독할 수 있는 특성을 이용한 것으로 일반적으로 컴퓨터 프로그램이 해독하기 어려운 기호, 글자 등을 재입력하도록 하여 스팸을 위한 자동화 도구 등을 무력화 시키는 보안 기술이다. 전 보고서에서 서브 이미지를 이용한 CAPTCHA 시스템을 제공하였다. 이미지에서 추출한 서브 이미지를 이용하여 사용자에게는 쉽고, 컴퓨터 프로그램에게는 어려운 CAPTCHA 시스템을 제공 할 수가 있었다. 전 보고서에는 우리는 사용자에게 보다 쉬운 이미지를 제공하기 위하여 이미지 내에 존재하는 컬러 색상을 이용하여 필터링을 제공할 수 있었다. 하지만 이미지에 존재하는 컬러 색상을 이용한 필터링만으로는 사용자에게 보다 쉬운 이미지를 제공할 수가 없었다. 본 보고서에서는 전 보고서에서 보여주었던 필터링 기능에 대하여 좀 더 논의하고, 실험을 통하여 사용자에게는 통과하기 쉬운 CAPTCHA 시스템을 제공할 수 있도록 제안한다.

KEYWORDS CAPTCHA, Sub-Image

1 서론

인터넷의 도래로 많은 웹 페이지 생성되고 해당 웹 페이지에서는 많은 서비스를 제공하고 있다. 많은 웹 페이지가 생성된 만큼 해당 웹 페이지에서 많은 서비스가 제공되고 있는 만큼, 사용자 또한 기하급수적으로 늘어가고 있다. 사용자가 많이 늘어나는 만큼 해당 사용자들이 이용하고 있는 홈페이지에서도 많은 정보들이 오가고 또한 사용자들 간의 교류도 활발히 일어 나고 있다. 이러한 교류 또는 정보 교환을 위하여 사용자들을 게시판을 이용한다. 하지만 사용자들은 가끔 게시판에 음란성 또는 광고성 글들이 무분별하게 작성되어 있는 것을 볼 수가 있다. 이러한 무분별하게 작성할수 있는 경우는 웹봇(Web-bot)을 이용하여 자동으로 글을 작성하는 것이다.

이러한 무분별한 정보기입을 막기 위해 사용자 등록 또는 게시판에 정보를 기입할 경우, 웹봇과 실제 사용자를 식별하기 위하여 CAPTCHA 시스템을 도입하였다. CAPTCHA 시스템의 기본적인 목적은 인터넷을 이용하는 실 사용자를 가리는 시스템이다. 즉, 인간의 우수한 가독성을 이용하여 인터넷을 이용하는 실 사용자를 컴퓨터와 인간으로 구분짓는 것이다. 현재 CAPTCHA 시스템은 크게 2 가지로 구분 지을 수 있다. 문자를 이용한 CAPTCHA 시스템과 이미지를 이용한 이미지 기반의 CAPTCHA 시스템으로 구분 지을 수 있다. 하지만 기존에 존재하고 있는 문자 기반의 CAPTCHA 시스템은 이미지 프로세싱, 머신 러닝 등 기존에 존재하고 있는 알고리즘을 통하여 쉽게 통과 될수 있다는 것을 알수가 있다. 문자 기반 CAPTCHA 시스템에 대하여 몇가지 알아보고 또한 해당하는 시스템이 어떠한 방법으로 쉽게 통과되는지에 대해서도 알아보도록 하자.

문자 기반 CAPTCHA 시스템중 HIP(Human Interaction Proof) [2] 방식은 텍스트 기반의 시스템이었다. 하지만 텍스트 기반의 CAPTCHA, HIP 시스템들은 머신러닝 [6] 을 통하여 쉽게 통과될 수 있는 것으로 알려졌다. 그림을 통하여 기존에 존재하는 문자 기반 CAPTCHA 시스템을 알아보도록 하자. 그림 1 는 기존에 존재하고

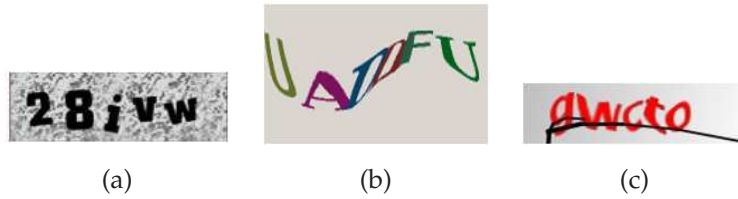


그림 1. 가장 일반적인 글자를 이용한 CAPTCHA 시스템. a) 와 같은 경우 너무 명확한 문자로 제공되어서 이미지 프로세싱 또는 문자인식을 통하여 쉽게 통과 할 수 있다. b)와 같은 경우 컴퓨터 프로그램에게는 어려움을 제공할 수 있는 문제이기는 하나 사용자에게는 다소 모호한 문자 기반 CAPTCHA 시스템을 제공하여 CAPTCHA 시스템의 기본적인 취지를 벗어나는 경향을 띄고 있다. c)와 같은 경우 사용자에게는 다소 쉬운 CAPTCHA를 제공하며 동시에 컴퓨터 프로그램에게는 다소 어려운 문제를 제공하나 Projection Based Algorithm 을 통하여 c)에 존재하는 검은 선을 제거하면 a)와 같은 기본적인 CAPTCHA 시스템이므로 컴퓨터가 쉽게 통과 할수 있다.

있는 문자 기반의 CAPTCHA 시스템들이다. 그림 1 중에 a) 와 같은 시스템은 가장 일반적인 글자를 이용한 CAPTCHA 시스템이라 할 수 있다. a) 와 같은 경우는 너무 명확한 문자로 제공되어서 이미지 프로세싱 또는 문자인식을 통하여 쉽게 통과 할 수 있는 것으로 알려졌다[6]. b) 와 같은 경우는 컴퓨터 프로그램에게는 어려움을 제공할 수 있으나 사용자에게는 다소 문자의 구분을 모호하게 만드는 경향을 띄어 CAPTCHA 기본적인 취지를 벗어나는 경우를 알 수 있다. c) 와 같은 경우는 사용자에게는 모호한 경우도 발생 시키지 않으며 인간의 우수한 가독성을 이용하여 효율적인 CAPTCHA 시스템을 제공하였으나, 기존에 존재하고 있는 알고리즘(Projection Based Algorithm) 을 이용하면[3] c)에 존재하고 있는 붉은 선을 쉽게 제거 할 수가 있다. 문자를 덮고 있는 선을 제거 할 경우 a)와 같은 시스템과 동일하므로 쉽게 CAPTCHA 시스템을 통과 할 수 있다. 이처럼 문자 기반을 이용한 CAPTCHA 시스템은 사용자에게는 쉬운 시스템을 제공하면, 컴퓨터 프로그램이 쉽게 통과 할 수 있게 되어 버리고 컴퓨터 프로그램에게 어려운 시스템을 제공하면 사용자에게는 모호한 환경 또는 어려움을 제공해버리는 결과를 초래하게 되었다. 이러한 문자 기반 시스템의 문제점을 파악하여 이미지 기반을 기초로 한 새로운 CAPTCHA 시스템이 제공 되었다[4]. 이미지 기반의 CAPTCHA 시스템은 특정한 개체를



(a)상하의 구분이 명확하지 않는 사진 (b)상하의 구분이 명확한 사진

그림 2. 이미지 기반의 CAPTCHA 시스템 중에 하나이다[4]. 이 이미지 기반의 CAPTCHA 시스템은 원 안에 이미지를 담아서 무작위 회전을 가한 후 사용자에게 교정을 요하는 시스템이다. 인간의 우수한 가독성을 이용한 이미지 기반의 CAPTCHA 시스템이나 이미지의 상하 구분을 정할 수 없는 이미지의 경우 사용자에게 어려움을 초래하게 된다. a) 와 같은 기타의 경우 교정을 가할 수 있지만, 어떠한 방향을 정답으로 간주해야 될지 어려운 상황이다. b)와 같은 자동차의 경우 사람은 정확하게 이미지 교정을 가할 수 있다.

선별해내는 시스템 또는 이미지에 올바른 교정을 요하는 시스템으로 나뉘어져 있다. 그림 2와 같은 경우는 사용자에게 올바른 이미지 교정을 통한 실 사용자를 인증하는 시스템이다. 하지만 그림 2에 있는 이미지 교정을

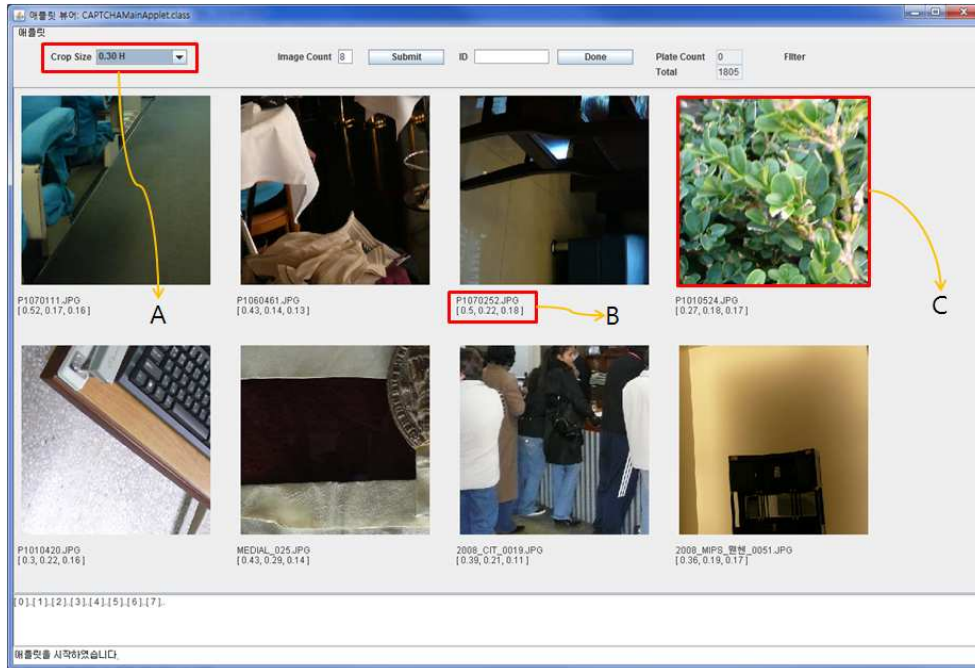
요하는 시스템은 그림 2에 존재하는 a)와 같이 교정을 할 수 있으나, 올바른 답으로 간주하기 어려운 이미지 또는 상하의 구분이 모호한 경우 사용자에게는 불편함을 초래한다. 이와 같은 시스템의 경우 컴퓨터에게는 다소 어려운 시스템일수도 있으나, 사용자에게는 모호한 경우를 제공 또한 상하의 구분이 없는 이미지를 제공해야 하므로 데이터의 수집이 제한적인것이 단점이다. 특정한 개체를 선별해내는 이미지 기반의 CAPTCHA 시스템은 이미지에 담겨진 개체의 의미를 알수 있는 특징을 이용한 시스템이다. [5]와 같은 시스템은 개와 고양이의 사진을 12장을 제공한 뒤, 12장의 사진에서 고양이의 사진을 선별해내는 시스템이다. 이 시스템에 임하는 실 사용자의 경우 99.6%의 확률로 통과할 수 있었다. 하지만 머신 러닝[11]를 통하여 82.7%로 시스템을 통과할 수 있었다.

이와 같이 현재 제공되고 있는 문자 기반 또는 이미지 기반의 CAPTCHA 시스템은 사용자에게 편의를 제고함과 동시에 컴퓨터 프로그램에게는 통과하기 어려운 시스템을 제공한다고 생각 하였으나 머신 러닝, 문자 인식 그리고 이미지 프로세싱 등 알고리즘을 통하여 통과 될 수 있는 것을 알 수 있다.

2 서브 이미지를 이용한 CAPTCHA

본 단락에서는 본 보고서에서 제안하는 이미지 기반의 새로운 CAPTCHA 시스템의 구조 및 시스템을 소개한다. CAPTCHA 시스템이 가져야 할 기본적인 요소는 자동생성 및 데이터의 다양성, 간편한 User Interface와 쉬운 난이도 그리고 웹봇 방지이다. 즉, 실사용자를 쉽게 판별 할 수 있는 사용자는 통과 하기 쉬운 CAPTCHA 시스템을 제공하며, 컴퓨터에게는 어려운 CAPTCHA 시스템을 제공하는 것이다. 문자 기반의 CAPTCHA 시스템은 문자의 조합으로 자동 생성과 다양한 데이터를 만들 수 있다. 그러나 이미지 기반 CAPTCHA[15] 시스템에서는 이러한 조합에 의한 생성이 불가능하기 때문에 많은 데이터의 확보가 중요하다. 본 논문에서 제안하는 CAPTCHA 시스템은 기존의 방식과는 달리 일반 사진의 일부분 즉, 전체 이미지가 아닌 부분 이미지(Sub-Image)를 이용한다. 서브 이미지를 이용할 경우 한 장의 이미지에서도 각기 서로 다른 서브 이미지를 추출할 수 있으므로 적은 양의 데이터로써도 충분한 CAPTCHA 시스템 환경을 제공할 수 있다.

본 보고서에서 제시하는 서브 이미지를 이용한 CAPTCHA 시스템의 기본적인 아이디어는 일반적으로 촬영된 이미지에서 무작위로 서브 이미지를 추출한 뒤 해당 이미지에 무작위 회전을 가한뒤 사용자에게 이미지 교정을 요하는 것이다. 서브 이미지를 추출하는 형태는 사각형의 형태를 이루며, 임의로 회전하는 방향은 0도, 90도, 180도 그리고 270도 이다. 그림 3은 본 보고서에서 제공되는 서브 이미지를 이용한 CAPTCHA 시스템이다. 본 보고서에서 제공되는 CAPTCHA 시스템은 총 8장의 임의의 사진에서 임의의 위치로 서브 이미지를 추출하여 각각의 서브 이미지에 무작위 회전을 가하여 CAPTCHA 시스템이 제공된다. 일반적으로 촬영된 이미지에서 서브 이미지 추출시 추출하는 크기는 원 이미지의 크기에 비례하여 추출된다. 본 보고서에서 제공되는 CAPTCHA 시스템은 사용자가 CAPTCHA 시스템에 임하여 사진의 교정을 쉽게 하기 위하여 각 서브 이미지의 크기 만큼의 영역을 할당하여 마우스 클릭으로 이미지 교정을 할수 있게 하였다. 그림 3에 있는 Combo Box A는 원 이미지에서 서브 이미지 추출시 크기를 변경시킬수 있는 인터페이스로 사용 된다. 그림 3에 있는 Box B는 서브 이미지내에 픽셀 값들을 하나씩 가져와 픽셀 값이 가지는 R, G, B 값들과 테이블 1에 있는 모든 색상들과 비교하여 가장 가까운 값으로 변경 시키는 작업을 통하여, 서브 이미지 값들을 전부 변경시킨다. 테이블 1의 색상 값들은 사람이 가장 쉽게 인지할 수 있는 색상 값들이다. 서브 이미지의 한 픽셀에 존재하는 R, G, B 값과 테이블 1에 존재하는 색상들을 수식 $(p_r - C_{ir})^2 + (p_g - C_{ig})^2 + (p_b - C_{ib})^2$ 을 통하여 모든 값들을 구한 뒤 최소 값을 가지는 컬러 색상 즉, 테이블 1에 있는 컬러 값으로 변경하여 서브 이미지의 전체 픽셀에 R, G, B 값들을 변경한다. 변경된 서브 이미지내에 색상 분포도를 조사하여 가장 높은 값 3가지를 표시한 결과가 바로 그림 3에 있는 Box B 이다. 사용자의 쉬운 이미지 교정을 위하여 각 서브 이미지의 영역안에서 마우스 클릭을 통하여 쉬운 이미지 교정을 할수가 있다. 본 단락에서는 본 보고서에서 제공되는 서브 이미지를 이용한 CAPTCHA



(a)본 보고서에서 제공되는 서버 이미지를 이용한 CAPTCHA 시스템

그림 3. 본 보고서에서 제공하는 서버 이미지를 이용한 CAPTCHA 시스템이다. 사용자는 Combo Box A 를 이용하여 원 이미지에서 추출되는 서버 이미지의 크기를 조정할 수 있다. Box B에 보이는 소숫점 자릿수 3개는 서버 이미지에 존재하는 한 픽셀 안에 있는 R, G, B 값들과 테이블 1 안에 있는 색상들과의 거리중 최소값을 가지는 색상값 즉, 최소값을 가지는 테이블 내의 색상값으로 대치하여 색상 분포도를 단일화 시킨 후 서버 이미지내에 색상 분포도를 파악하여 가장 큰 값을 오름 차순으로 정렬하여 가장 높은 값을 가지는 3개의 값을 보여주고 있는 것이다. 제공되는 CAPTCHA 시스템은 Box C와 같은 크기로 서버 이미지가 제공되며, 사용자는 각각의 서버 이미지의 영역안에서 마우스의 클릭으로 쉽게 이미지 교정을 할 수 있다.

시스템을 알아보았다.

3 실험

본 단락에서는 본 보고서에서 제시한 서버 이미지를 이용한 이미지 기반의 CAPTCHA 시스템을 실험을 통하여 사용자에게 제공하였을 때에 확률적으로 성공률이 가장 높은 서버 이미지의 비율을 찾고 색상 분포도에 따른 사용자의 성공률에 대해서도 알아보겠다. 본 실험은 최고 200 명의 실험자들을 통해 실험에 임하였다. 실험의 전반적인 배경은 총 8장의 서버 이미지를 제공하여 각각의 이미지에 대하여 성공 또는 실패의 여부를 따지어 8개의 서버 이미지를 제공한 뒤 7개의 서버 이미지에 대하여 올바른 이미지 교정을 했을 경우, 서버 이미지 CAPTCHA 시스템에 통과한 것으로 간주하였다.

본 단락에서의 실험의 결과는 테이블 2와 같다. 테이블의 성공률은 8개의 서버 이미지에 대하여 7개이상의 서버 이미지에 대하여 올바른 교정을 했을 경우 본 CAPTCHA 시스템을 통과했다는 거와는 달리 테이블 2은 각각의 서버 이미지에 대한 성공률을 보여주는 것이다. 테이블 2 에 보이는 값들을 보면 서버 이미지의 비율이 낮을 경우 사용자는 서버 이미지에 대한 의미를 찾기 힘들어 많은 클릭을 통하여 교정에 임하는 것을 알수 가

Color	R	G	B
Black	0	0	0
Sea Green	0	182	0
Light Green	0	255	179
Olive Green	36	73	0
Aqua	36	146	170
Bright green	36	255	0
Blue	73	36	170
Green	73	146	0
Turquoise	73	219	170
Brown	109	36	0
Blue gray	109	109	170
Lime	109	219	0
Lavender	146	0	170
Plum	146	109	0
Teal	146	182	170
Dark red	182	0	0
Magenta	182	73	170
Yellow green	182	182	0
Flouro green	182	255	170
Red	219	73	0
Rose	219	146	170
Yellow	219	255	0
Pink	255	36	170
Orange	255	146	0
White	255	255	255

표 1. 인간이 가장 인지하기 쉬운 색상 25 가지. 이 색상들을 이용하여 서브 이미지내에 각 픽셀에 존재하는 R, G, B 값의 거리를 구한 뒤 최소값을 가지는 색상을 구하여 서브 이미지의 색상 값을 변경시켜 서브 이미지 내에 색상들을 단일화 시킨다.

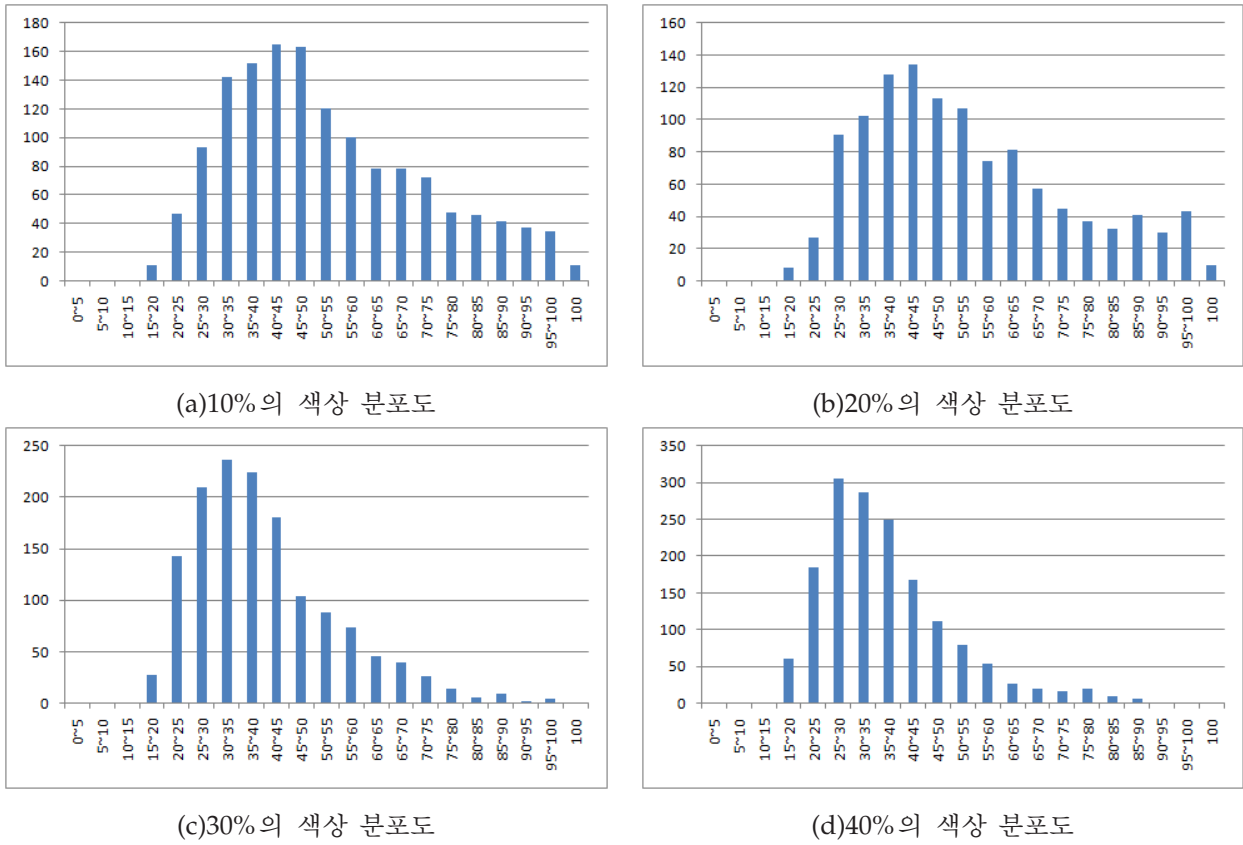


그림 4. 서버 이미지의 비율에 따른 색상 분포도를 보여 주고 있다. Y축은 색상에 따른 갯수를 따르고 있으며, X축은 이미지에 대한 색상 분포도를 나타내고 있다.

있으며, 서버 이미지에 대한 의미가 낮게 부여된 만큼 정답률도 저조한 것으로 나타났다. 하지만 서버 이미지의 비율이 증가 됨에 따라 의미가 조금씩 부여되며, 정답률도 올라가며 의미가 부여됨에 따라 사용자는 이미지에 대한 올바른 교정을 위해 지속적인 마우스 클릭을 통하여 이미지 회전을 한 것으로 보인다. 서버 이미지의 비율이 25%에서는 가장 많은 클릭을 보였으며, 정답률의 증가도 가장 높은 것으로 나타났다. 하지만 25%에 비하여 30%에서는 사용자들의 클릭수는 25%에 비하여 낮아진 것을 알수 가 있다. 이것은 빠른 교정을 통해 더욱더 사용자에게 쉬운 환경을 제공하였다고 가정하여, 30%가 사용자에게 가장 좋은 서버 이미지 추출할수 있는 비율이라고 가정하였다.

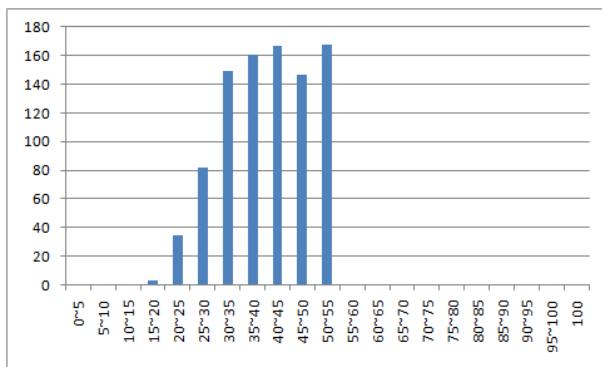
또한 본 단락에서는 앞 단락에서 논한 바가 있는 테이블 1을 이용한 서버 이미지의 색상을 단일화 시키는 방법에 대하여 논한 바가 있었다. 본 단락에서는 테이블 1를 이용하여 서버 이미지의 색상 분포도를 각 서버 이미지의 추출 비율에 따라 살펴보고, 또한 색상 분포도에 있어서 한 가지의 색상이 너무 많은 비율을 차지 하고 있으면, 해당 서버 이미지를 CAPTCHA 시스템에서 제외시키고 이와 같은 필터링을 통하여 사용자의 성공률은 얼마나 증가되는지에 대하여 알아보겠다.

그림 4을 살펴보자. 그림 에서 a)의 그래프는 서버 이미지의 추출시 원 이미지에서 10% 비율로 서버 이미지를 추출한 것이다. 서버 이미지의 비율이 낮고 또한 서버 이미지의 비율에 낮음에 따라서 색상 분포도도 균일 하지 못한 것을 알수가 있다. 즉, 40% 이상 45%미만 또는 45% 이상 50%미만이 많은 부분을 차지 하고 있다. 따라서 한 서버 이미지에서 한 가지의 색상이 거의 50%을 차지한다는 것이다.. b) 에 해당하는 그래프는 서버 이미지의 추출시 원 이미지에서 20%의 비율로 추출한 것이다. 20%의 추출시 색상분포도는 10%와 비슷하다.

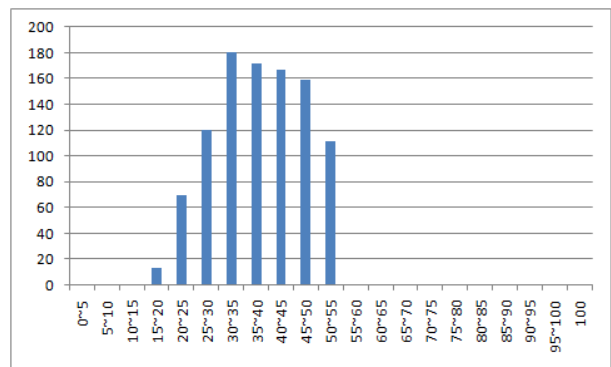
서브 이미지의 비율	총 사진 갯수	정답률	사용자 클릭수	평균 클릭수	평균 시간
10%	832	76.40%	674	0.81	6
15%	832	84.13%	935	1.12	10
20%	824	86.77%	948	1.15	14
25%	832	91.58%	1215	1.46	16
30%	816	93.62%	1155	1.41	13
35%	816	95.00%	932	1.14	9
40%	832	97.00%	926	1.11	6
45%	832	98.90%	632	0.75	5

표 2. 서브 이미지 하나에 대한 사용자들의 성공률, 정답률 및 마우스 클릭수를 보여주고 있는 테이블 사용자들은 서브 이미지를 추출할 수 있는 비율에 적을 수록 서브 이미지에 대한 의미를 찾기 힘들어서 평균 클릭수가 낮고 성공률이 낮은 반면, 서브 이미지의 비율이 증가됨에 따라 의미가 부여되어 점점 클릭수가 증가되며 30%에서 부터는 사용자 클릭수가 감소하는 양상을 띄어 30%의 서브 이미지가 사용자에게 가장 의미 있는 사진을 부여할수 있다고 가정한다. 클릭수가 감소하는 양상은 이미지에 의미가 부각되어서 쉬운 교정을 할 수 있다는 것으로 간주할 수 있다.

가장 높은 수치를 나타내는 부분이 10%와 20%이 같은 수치를 나타내고 있다. 하지만 20%의 경우 10%보다 조금 더 색상 분포다가 균일한 것으로 나타난다. 하지만 30%의 경우 10%, 20%의 경우와는 다르게 가장 높은 수치를 나타내는 부분이 다르게 나타났다. 고로 서브 이미지를 추출할 경우 이미지의 비율이 즉, 서브 이미지의 크기가 증가될수록 색상분포도가 균일한 것으로 나타났다. 40%역시 가장 높은 수치를 나타내는 곳이 30%와 동일하게 나타내었으나 전체적인 분포도가 40%가 더 높은 곳으로 나타났다. 즉, 이미지의 비율이 증가될수록 이미지의 크기가 증가될수록 색상 분포도가 균일하게 나타나는 것으로 나타났다. 하지만 이 실험은 각 서브 이미지의 사이즈마다 250 장을 추출하여 평균적인 분포도를 나타낸것이다. 즉, 30% - 40%경우에도 색상 분포도가 균일하게 나타나지 않는 서브 이미지 도 존재하는 것으로 나타났다. 색상 분포도가 균일하지 못하고 한 가지의 색상이 70% 되는 서브 이미지의 경우 사용자에게 모호함을 일으킨다. 그리하여 본 실험에서 색상 분포도가 균일 하지 못한 서브 이미지에 대하여 필터링을 제공하고 필터링 시킨 CAPTCHA 시스템에 대하여 실험을 해보았다.



(a)10%의 색상 분포도



(b)20%의 색상 분포도

그림 5. 서브 이미지내에서 한 가지의 색상 분포도가 55% 이상이 되는 서브 이미지를 제외 시킨 후, 평균 분포도를 나타내는 그래프

그림 5는 서브 이미지내에 존재하는 색상 분포도중에 한 가지의 색상이 55% 차지할 경우 서브 이미지를 이용한 CAPTCHA 시스템에서 제외 시켰다. 우리는 이러한 필터링 시스템을 통하여 200 명의 사용자들을 통하여 실험에 임하였다. 그림 5에 a) 의 그래프는 서브 이미지의 비율 10%로 추출하여 색상 분포도를 나타낸 것이고, 20%의 경우 b) 에 해당한다. 추출된 그래프를 살펴보면 필터링을 통하여 색상 분포도는 달라진 것은 없으나 테이블 을 통하여 성공률이 더 증가된 것을 알수가 있다. 즉, 서브 이미지 추출시 필터링을 통하여 한 가지의 색상이 너무 많이 차지 하는 이미지를 제외시켰을 경우 필터링을 제외시킨 경우 보다 성공률이 더 올라간 것으로 판단된다.

4 결론

본 보고서에서는 CAPTCHA 시스템에 제공될 가장 알맞은 서브 이미지의 비율을 찾기 위하여 200 명의 실험자들을 통한 실험을 행하였다. 실험을 통하여 서브 이미지의 비율이 높아짐에 따라 높은 성공률을 보였다. 높은 성공률을 보이며 사용자들이 올바른 이미지 교정을 위하여 가장 많은 클릭을 보였던 25% 이상인 30% 를 가장 이상적인 서브 이미지의 비율로 정하였다. 또한 본 보고서에서는 이미지의 비율이 아닌 추출된 서브 이미지에 의미가 부여되어 사용자가 쉽게 CAPTCHA 시스템에 임할수 있도록 테이블 1 을 이용하여 색상을 단일화 시킨 뒤, 서브 이미지에서 한 가지의 55% 이상을 차지하는 서브 이미지는 필터링을 통하여 CAPTCHA 시스템에서 제공하지 않았다. 또한 이 필터링을 제공한 뒤, 200 명의 실험자들을 통하여 실험을 행하였다. 필터링을 통한 CAPTCHA 시스템에서는 더 높은 성공률을 나타내었다. 하지만 필터링을 행한다 하여도 한 가지의 색상이 40% 정도 차지하여도 사용자에게는 많은 어려움을 초래하는 것으로 나타났다. 그리하여 다른 방법으로 한 서브 이미지를 25 가지 색상으로 단일화 한 뒤, 25 가지의 색상들이 고르게 분포되어 있는 서브 이미지를 추출해야 할것이다. 즉, 테이블 1 을 통하여 서브 이미지의 색상을 단일화 시킨뒤, 25 가지의 색상을 많이 고르게 가지고 있는 이미지를 추출하는 것이다. 우리는 다음과 같은 필터링을 통하여 또 한번의 실험을 거쳐서 사용자에게는 보다 더 낫은 CAPTCHA 시스템을 제공할 수 있을 것이다.

참고 문헌

1. Shumeet Baluja, Automated image-orientation detection: a scalable boosting approach, *Pattern anal. appl.* **10** (2007), 247-263.
2. Kumar Chellapilla, Kevin Larson, Patrice Simard, and Mary Czerwinski, Designing human friendly human interaction proofs (HIPs), Proc. of the SIGCHI conference on Human factors in computing systems, 2005, pp. 711-720.
3. Huang SY et al., A projection-based segmentation algorithm for breaking MSN and YAHOO CAPTCHAs, Proc. of the International Conference of Signal and Image Engineering, 2008.
4. Rich Gossweiler, Maryam Kamvar, and Shumeet Baluja, What's up CAPTCHA?: a CAPTCHA based on image orientation, Proc. of the 18th international conference on World wide web, 2009, pp. 841-850.
5. J. Howell J. Elson, JR. Douceur and J. Saul, Asirra: a CAPTCHA that exploits interest-aligned manual image categorization, Proc. of the 14th ACM conference on Computer and communications security, 2007, pp. 366-374.
6. P Simard K Chellapilla, Using machine learning to break visual human interaction proofs (HIPs), Advances in Neural Information Processing Systems, 2004.
7. Hong-Jiang Zhang Lei Zhang, Mingjing Li, Boosting Image Orientation Detection with Indoor vs. Outdoor Classification, Proc. of the Sixth IEEE Workshop on Applications of Computer Vision, 2002, pp. 95-99.
8. David G. Lowe, Distinctive Image Features from Scale-Invariant Keypoints, *International journal of computer vision* **60** (2004), 91-110.
9. Jiebo Luo and Matthew Boutell, A Probabilistic Approach to Image Orientation Detection via Confidence-Based Integration of Low-Level and Semantic Cues, CVPRW '04: Proceedings of the 2004 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'04) Volume 9, IEEE Computer Society, 2004, p. 141.

10. G. Mori and J. Malik, Recognizing objects in adversarial clutter: breaking a visual CAPTCHA, Proc. of Computer Vision and Pattern Recognition., June 2003.
11. Golle Philippe, Machine learning attacks against the Asirra CAPTCHA, Proc. of the 15th ACM conference on Computer and communications security, 2008, pp. 535–542.
12. B. G. Prasad, K. K. Biswas, and S. K. Gupta, Region-based image retrieval using integrated color, shape, and location index, *Comput. vis. image underst.* **94** (2004), 193–233.
13. et al. S. M. Bileschi, Towards component-based car detection, In ECCV Workshop on Statistical Learning and ComputerVision, 2004.
14. Lyu Siwei, Automatic image orientation determination with natural image statistics, Proc. of the 13th annual ACM international conference on Multimedia, 2005, pp. 491–494.
15. V. Srikanth, C. Vishwanathan, Udit Asati, and N. Ch. Sriman Narayana Iyengar, Think-an image based CAPTCHA mechanism (testifying human based on intelligence and knowledge), Proc. of the International Conference on Advances in Computing, Communication and Control, 2009, pp. 421–424.
16. A. Vailaya, H. Zhang, Changjiang Yang, Feng-I Liu, and A.K. Jain, Automatic image orientation detection, *IEEE trans. on image processing* **11** (2002), 746–755.
17. Lei Wang, Xu Liu, Lirong Xia, Guangyou Xu, and Alfred Bruckstein, Image orientation detection with integrated human perception cues (or which way is up), Proc. of ICIP, vol. 2, Sept. 2003, pp. 539–42.
18. Yongmei Wang and Hongjiang Zhang, Content-based image orientation detection with support vector machines, IEEE Workshop on Content-Based Access of Image and Video Libraries, 2001.
19. Yongmei Michelle Wang and Hongjiang Zhang, Detecting image orientation based on low-level visual content, *Computer vision and image understanding* **93** (2004), 328 – 346.
20. Ming-Hsuan Yang, David J. Kriegman, and Narendra Ahuja, Detecting Faces in Images: A Survey, *IEEE transactions on pattern analysis and machine intelligence* **24** (2002), 34–58.