Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.25, NO.5, Oct. 2015 ISSN 2288-2715(Online)
http://dx.doi.org/10.13089/JKIISC.2015.25.5.1011

특징 분리를 통한 자연 배경을 지닌 글자 기반 CAPTCHA 공격*

김 재 환,† 김 수 아, 김 형 중‡
고려대학교 정보보호대학원

Breaking character and natural image based CAPTCHA using feature classification*

Jaehwan Kim,† Suah Kim, Hyoung Joong Kim‡
Center for Information Security Technologies(CIST), Korea University

요 약

컴퓨터 사용자가 사람인지 아닌지를 판별하는 CAPTCHA는 많은 포털 사이트에서 자동 프로그램에 의한 비정상적인 회원가입 또는 다중 로그인 방지 등을 위해 사용되고 있다. 많은 웹 사이트들은 숫자 혹은 영어로 구성된 문자열 기반 캡챠를 대부분 사용하는데, 최근에는 OCR 기술의 발달로 단순한 텍스트 기반 캡챠는 쉽게 무력화 된다. 이에 대한 대안으로 많은 웹 사이트들은 글자 판독을 어렵게 하기 위해 잡음을 첨가하거나 글자를 왜곡시키는 등 다양한 시도를 하고 있다. 본 논문에서 대상으로 하는 국내 한 포털 사이트 역시 공격자들에 의해 많은 공격을 당하였고, 끊임없이 캡챠를 발전시키고 있다. 본 논문에서는 해당 사이트에서 현재 사용되고 있는 다양한 자연 배경을 지닌 캡챠에 대해 분석하고, SVM을 이용한 특징 분리 후 CNN을 이용한 글자 인식을 통해 해당 캡챠의 취약성을 검증하였다. 실험 결과, 총 1000개의 캡챠 이미지 중 368개에 대해 정확히 맞추었고, 이를 통해 해당 포털 사이트에서 현재 사용하고 있는 새로운 버전의 캡챠 역시 안전하지 않음을 입증하였다.

ABSTRACT

CAPTCHA(Completely Automated Public Turing test to tell Computers and Humans Apart) is a test used in computing to distinguish whether or not the user is computer or human. Many web sites mostly use the character-based CAPTCHA consisting of digits and characters. Recently, with the development of OCR technology, simple character-based CAPTCHA are broken quite easily. As an alternative, many web sites add noise to make it harder for recognition. In this paper, we analyzed the most recent CAPTCHA, which incorporates the addition of the natural images to obfuscate the characters. We proposed an efficient method using support vector machine to separate the characters from the background image and use convolutional neural network to recognize each characters. As a result, 368 out of 1000 CAPTCHAs were correctly identified, it was demonstrated that the current CAPTCHA is not safe.

Keywords: CAPTCHA, Breaking CAPTCHA, SVM, CNN, HSV color space

접수일(2015년 5월 4일), 수정일(1차: 2015년 7월 14일, 2차: 2015년 8월 3일), 게재확정일(2015년 8월 19일)

* This work was supported by the National Research Foundation of Korea Grant funded by the Korean government(MEST) (NRF-2015R1A2A2A01004587) and supported by Business for Cooperative R&D between Industry, Academy, and Research Institute funded Korea Small and Medium Business Administration in 2015 (C0191629)

† 주저자, edenkim519@korea.ac.kr
‡ 교신저자, khj-@korea.ac.kr(Corresponding author)

I. 서 론

CAPTCHA(Completely Automated Public Turing test to tell Computers and Humans Apart)는 사용자가 사람인지 컴퓨터인지를 판단하기 위해 사용되는 기술의 일종으로, 많은 포털 사이트에서 회원가입 혹은 다중 로그인 방지 등을 위해 사용되고 있다[1].

가장 흔히 볼 수 있는 캡챠는 텍스트 기반 캡챠로, 숫자 혹은 영어로 구성되어 있는데, 최근에는 OCR(Optical Character Reader) 기술의 발달로 단순한 텍스트 기반 캡챠는 쉽게 공격이 가능하게 되었다. 이에 텍스트 기반 캡챠를 운영하는 많은 포털 사이트들은 글자를 쉽게 판독하지 못하도록 글자에 잡음을 넣거나, 기울이기, 붙이기 등 다양한 방법을 사용하여 글자의 판독을 어렵게 만든다. 이 외에도 간단한 연산 질문에 답을 하는 수학 캡챠[2], 음성 파일을 들려주는 오디오 기반 캡챠[3], 특정 사진을 보여주고 이에 관련된 질문에 답하는 이미지 기반 캡챠[4] 등 여러 종류의 캡챠들이 존재한다.

본 논문에서는 국내 인터넷 사용자의 80% 이상이 이용하는 한 포탈 사이트의 다양한 자연 배경 그림을 지닌 글자 기반 캡챠를 분석하고, 실험 결과를 통해 해당 캡챠가 보안에 취약함을 소개하고자 한다.

Ⅱ장에서는 공격 대상 캡챠의 동향 및 특징을 소개하고, Ⅲ장에서는 캡챠 분석 알고리즘에 대해 서술하며, Ⅳ장에서는 실험 결과를 서술하고, Ⅴ장에서는 결론으로 해당 논문을 마무리 짓는다.

II. 공격 대상 캡챠의 동향 및 특징

문자열 기반 캡챠는 현재까지 많은 포탈에서 사용되고 있으며, 이를 공격하기 위한 많은 연구들이 진행되었다[5]. 사실 캡챠를 공격하는 가장 중요한 단계는 글자 분리와 글자 인식 단계이다. 그런데 글자 인식의 경우 SVM(Support Vector Machine)[6], KNN (K Nearest Neighbors)[7], CNN (Convolutional Neural Network)[8] 등 다양한 분류기가 이미 존재하고, 글자 분리만 잘 된다면 해당 분류기를 통해 95%가 넘는 글자 인식율을 보인다. 그러므로 많은 포털 사이트들은 글자 분리가 어려운 캡챠를 만들기 위해 노력하고 있다.

2.1 공격 대상 캡챠의 동향

본 논문에서 대상으로 하는 한 포털 사이트는 국내 사용자의 80% 이상이 사용하는 곳으로 많은 사용자를 보유하고 있어 공격자들의 타겟이 집중되고 있다. 사실 해당 포털 사이트는 옛날부터 많은 공격을 당했다. 2009년 김성호 등[9]에 의해 색상 정보를 이용한 공격으로 당시의 캡챠가 안전하지 않음을 선보였으며, 이후 해당 포털 사이트는 Fig. 1.과 같은 기울어진 캡챠를 개발하여 사용하였다. 하지만 이 역시 2013년 양대헌 등[10]에 의해 93%가 넘는 성공률로 tilting 기반의 collapsing 캡챠가 취약함을 입증하였다.

Fig. 1. CAPTCHA of N portal site(Old version)

2.2 기존 공격 방법의 한계점

해당 포털사이트에 5회 이상 로그인 실패 시 Fig. 2.와 같은 자연 배경을 지닌 캡챠가 발생한다. 이 외에도 다양한 배경을 지닌 캡챠들이 존재하며 그 중 일부를 Fig. 3.에 나타내었다.

새로운 버전의 캡챠는 예전 캡챠와 방식이 많이 바뀌었다. 예전 버전의 캡챠는 글자가 기울어지거나 글자를 붙이는 등 사용자의 가독성이 떨어지는 방법

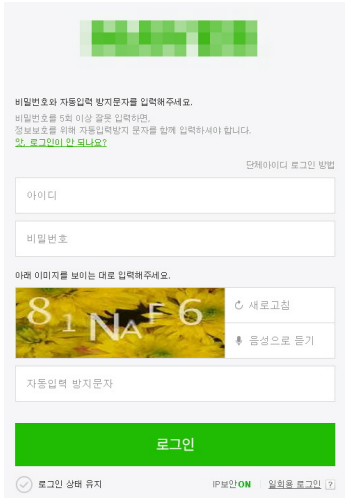


Fig. 2. Login page of N portal site

Fig. 3. CAPTCHA of N portal site (Updated version)

을 사용하여 글자 분리를 어렵게 만들었지만 새로운 버전의 캡챠는 글자의 변형을 최소화하여 가독성을 높이는 대신 다양한 자연 이미지를 배경으로 삽입하여 글자 추출 자체를 어렵게 만들었다.

즉 기존의 [9][10]에서 제시한 방법은 자연 배경이 없어 쉽게 글자를 추출할 수 있었기에 가능한 공격이었다. 새로운 캡챠는 배경으로부터 글자 추출 자체가 힘들어졌기 때문에 기존 방법으로는 공격이 불가하며 이를 위한 새로운 알고리즘을 본 논문 Ⅲ장에서 기술한다.

2.3 새로운 캡챠의 특징 (`15.06월 기준)

해당 포털 사이트에서 현재 사용되고 있는 캡챠의 특징은 다음과 같다. Fig. 3.의 상위 4개 캡챠는 다중 로그인 방지를 위한 캡챠이며, 하위 4개 캡챠는 카페 가입 시 발생되는 캡챠이다.

글자 수는 최소 4자리에서 최대 8자리까지로 공통적이나 다중 로그인 방지를 위한 캡챠는 알파벳과 숫자로 구성되어 있고, 카페 가입을 위한 캡챠는 숫자로만 구성되어 있는 특징이 있다.

캡챠에 나타난 글자를 올바르게 입력하지 않았을 경우 이전 캡챠와 다른 배경 그림과 글자를 지닌 캡챠가 생성되며, 로그인 실패에 대한 특별한 재제가 없어 무제한으로 로그인 시도가 가능하다.

Ⅲ. 새로운 캡챠를 위한 공격 방법

해당 포털 사이트에서 사용하고 있는 자연 배경을 지닌 캡챠의 효과적인 공격을 위한 알고리즘 순서도는 Fig. 4.와 같다. 해당 캡챠는 다양한 색상의 배경을 지녀 글자에 해당하는 부분의 컬러 범위가 캡챠마다 조금씩 다르다. 즉 고정 스레쉬홀드를 이용한 글자 추출 방법은 모든 캡챠에 대해 적용하기에 한계가 있다. 이에 본 장에서 제시하는 새로운 캡챠를 위한 공격 방법은 캡챠의 배경 정보에 따라 컬러 범위를 동적으로 조절해 최적의 글자를 분리하는 방법을 사용한다.

순서도를 간단히 살펴보면, 1단계로 분석할 캡챠를 읽은 뒤, 2단계로 SVM에서 사용할 특징을 추출한다. 3단계에서는 2단계에서 추출한 포그라운드 정보와 백그라운드 정보를 이용하여 SVM을 통해 글자에 해당하는 픽셀을 분류한다. 4단계에서는 추출한 글자들을 하나씩 분리하며 마지막 5단계에서는 CNN을 이용하여 분리한 각각의 글자들을 인식한다.

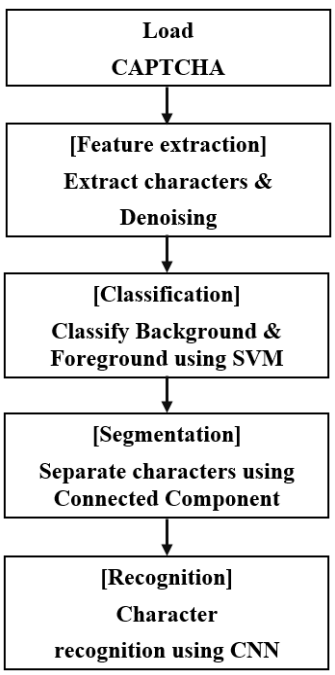


Fig. 4. Algorithm Flow

3.1 특징 추출 및 잡음 제거

3.1.1 포그라운드 정보 추출 및 잡음 제거

해당 캡챠의 효과적인 글자 분리를 위해서는 글자에 해당하는 포그라운드 정보를 잘 추출하는 것이 중요하다. 이를 위해 밝기 정도를 이용해 글자를 추출한다. 밝기를 구하는 방법은 수식 (1)과 같다.

$$Intensity = (R+G+B)/3 \quad (1)$$

글자 추출을 한 뒤, 글자 부분이 아닌 다른 곳에서 예상치 못한 잡음이 발생할 수 있는데, 이는 잡음 제거 과정을 통해 해결한다. 잡음 제거는 거리 기반 군집화를 사용하여 군집되어 있는 픽셀들의 개수가 60개 이하일 경우 이를 잡음으로 생각하고 해당 군집을 제거한다. 캡챠 분석 결과, 한 글자는 평균적으로 60개 이상의 픽셀이 군집되어 있음을 확인하였고, 이에 60개 이하의 픽셀을 가진 군집들은 잡음으로 간주하였다.

여기서 중요한 점으로 해당 캡챠는 배경 그림의 밝기가 어두운 것부터 밝은 것까지 다양하므로 적절한 값의 밝기 스레쉬홀드를 정해야만 잡음이 없는 정확한 글자 추출이 가능하다. 만약 낮은 스레쉬홀드 범위를 주게 되면 Fig. 6.와 같이 글자는 모두 추출되나, 많은 잡음이 발생하여 효과적인 글자 분리가 어렵고, 높은 스레쉬홀드 범위를 주게 되면, 밝기 신호가 약한 글자들은 잡음 제거 과정을 통해 Fig. 7.과 같이 글자의 부분들이 사라질 수 있다.

본 논문에서 제안하는 알고리즘에서는 83% 이상의 높은 스레쉬홀드를 주어 글자의 부분들이 조금 사라지더라도 최대한 잡음이 없도록 설계하였다. 잡음이 발생한 포그라운드 정보는 3.2절의 SVM을 이용한 특징 분리 과정에서 좋지 않은 결과를 초래한다. 83% 스레쉬홀드 범위에 포함되지 못하였거나 혹은 포함되었어도 잡음으로 간주되어 사라진 글자의 부분들은 3.2절의 특징 분리 과정을 통해 복원이 가능하다.

3.1.2 백그라운드 정보 추출

3.2절의 특징 분리에 사용될 파라미터로 포그라운드 정보 외에도 백그라운드 정보를 사용한다. 백그라운드 정보는 Fig. 8.과 같이 캡챠의 가장자리 부분을 이용한다. 캡챠의 글자는 항상 중앙 부분에 위치하는 특성이 있기 때문에 캡챠의 가장자리에 해당하는 부분들은 배경 그림에 해당하는 백그라운드 정보라고 가정할 수 있다.

Fig. 5. Original CAPTCHA

Fig. 6. Bad result (using low threshold)

Fig. 7. Foreground of CAPTCHA

Fig. 8. Background of CAPTCHA

3.2 SVM을 이용한 백그라운드와 포그라운드 특징 분리

해당 절에서는 3.1절에서 추출한 포그라운드와 백그라운드 정보의 특징 분리를 통해 캡챠의 배경 정보에 따라 글자에 해당하는 컬러 범위를 동적으로 조절해 글자를 효과적으로 분리한다.

특징 분리를 위한 분류기로는 SVM을 사용한다. SVM 분류기는 글자와 배경 특징 사이 최대한의 여백(margin)을 찾아 초평면(hyper-plane)을 생성해주기 때문에 타 분류기에 비해 일반화 능력이 뛰어난 특징이 있다. 파라미터로는 영상의 컬러 공간을

표현하는 모델 중 하나인 HSV color space 중 Saturation(채도), Value(명도) 그리고 영상의 Texture feature인 LBP(Local Binary Pattern) 정보를 사용한다. 위 파라미터들은 포그라운드와 백그라운드 정보를 분리할 좋은 특징을 지니고 있으며 보다 자세한 설명은 아래 기술한다.

3.2.1 Color space (HSV channel)

HSV color 모델은 인간의 색 인지에 기반을 둔 모델로 색 인지 방식이 직관적이기 때문에 컴퓨터 비젼 분야에서 많이 사용되는 색 공간 모델이다[11].

HSV color 모델은 색상을 나타내는 Hue, 채도를 나타내는 Saturation, 밝기를 나타내는 Value로 이루어져있는데, 본 알고리즘에서는 Saturation과 Value 정보 2가지를 이용해 글자 분리에 사용한다. 글자에 해당하는 하얀 부분은 Saturation이 낮고, Value가 높은 특징이 있다. Hue 정보는 글자와 배경을 분리할 큰 특징을 지니고 있지 않아 사용하지 않는다.

Fig. 9.와 Fig. 10.은 Saturation과 Value 각각의 정보를 나타낸 그림으로 검정색에서 하얀색으로 갈수록 값이 높아짐을 의미한다. 그림에서 보이는 것처럼 글자에 해당하는 부분은 Saturation이 낮고, Value가 높은 특징이 있으며 위 2개의 파라미터는 글자와 배경 그림을 분리하는데 좋은 특징을 가진다.

본 논문에서는 직관적인 이해를 돕기 위해 배경과 글자간의 채도 차가 큰 붉은 계열의 배경을 지닌 캡챠를 이용하였고 이에 Fig. 9.와 같은 뚜렷한 글자 결과를 얻을 수 있었다. 만약 무채색에 가깝고 밝은 배경을 지닌 캡챠의 경우는 Fig. 9.와 같은 뚜렷한 특징을 얻을 수 없고 Value 정보와 Saturation 정보를 같이 이용해야만 효과적인 글자 추출이 가능하다.

Fig. 9. Saturation channel

Fig. 10. Value channel

3.2.2 LBP Texture

특징 분리를 위한 또 하나의 파라미터로 영상의 Texture feature를 나타내는 LBP 정보를 사용한다[12]. LBP는 영상의 텍스처를 표현하는 뛰어난 방법 중 하나로 특히 얼굴 인식 분야에서 많이 사용되며 영상의 밝기 변화와는 무관하게 영상 내 내부 패턴 변화를 표현할 수 있는 장점이 있다.

Fig. 11.은 LBP 결과로 글자에 해당하는 Texture 정보를 이용해 글자의 식별이 가능함을 알 수 있으며 이는 글자와 배경을 분리하는데 좋은 특징이 된다.

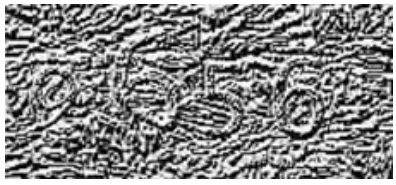


Fig. 11. LBP Texture

3.2.3 SVM Classifier

위에서 언급한 Saturation, Value, LBP 3가지 파라미터를 이용하여 SVM을 이용한 특징 분리를 진행하게 된다. 다만 본 논문에서는 SVM의 실행 결과를 그림으로 보이기 위해 Saturation과 Value 2가지 파라미터만을 이용하여 설명한다.

Fig. 12.는 SVM 분류 전 그림으로 좌측 상단의 별 표시(*)는 3.1절에서 추출한 포그라운드의 Saturation, Value 값을 나타낸다. 3.2.1절에서 확인했듯이 글자에 해당하는 하얀 부분은 Value가 높고 Saturation이 낮은 곳에 분포되어 있음을 알 수 있다. 그림 우측에 분포하는 플러스 표시(+)는 백그라운드에 해당하는 Saturation, Value 값을 나타낸다. 본 논문에서 예제로 보이고 있는 캡챠는 붉은 계열의 배경으로 가장자리에 해당하는 픽셀들은 Saturation이 높고 Value는 비교적 골고루 분포되어 있음을 알 수 있다. 위 학습 데이터를 이용해 SVM은 Support Vector를 정하고 백그라운드와

포그라운드 사이 최대 여백을 갖는 선(초평면)을 생성한다. 분류를 위한 커널로는 RBF(Radial Basis Function)를 사용하였다.

Fig. 13.은 분류 결과로 원본 캡챠(Fig. 5.)의 모든 픽셀들의 Saturation, Value 값을 표시하였다. 생성된 선을 기준으로 윗부분은 글자, 아랫부분은 배경으로 분류한다.

Fig. 14.는 분류 결과를 그림으로 나타낸 것으로, 왼쪽 그림은 SVM을 사용하기 전 학습 데이터(Fig. 12의 선 윗부분)이며 오른쪽 그림은 학습 후 분류된 데이터(Fig. 13의 선 윗부분)이다. 학습 후 글자에 해당하는 부분들이 정확히 추출되었음을 알 수 있고 이는 SVM 분류기가 배경과 글자를 잘 분리하였음을 의미한다.

3.3 글자 분리 및 CNN을 이용한 글자 인식

글자 인식을 위해 3.2 단계에서 얻은 결과로부터 글자를 각각 분리하여 글자를 인식하게 된다. 본 논문에서는 8-Neighborhood 기반 군집화 방법을 사용하여 글자를 분리한다.

분리된 각각의 글자들은 글자 인식 분류기를 통해 글자를 판별하게 되는데 본 알고리즘에서는 CNN을 사용하였다[13]. CNN은 글자 인식 분류기 중 현재까지 인식율이 가장 좋은 것으로 평가되고 있으며, CNN을 이용한 글자 인식 데모 영상으로는 [14]가 있다.

글자 인식은 Ⅱ장에서 언급했듯이 SVM, KNN, CNN 등 다양한 분류기가 이미 존재하고 인식율이 모두 뛰어나다. 이 중 인식율이 가장 뛰어나다고 평가되는 CNN 라이브러리를 본 논문에서 사용한 것이며 글자 인식은 본 논문의 주된 포커스가 아니기에 자세한 설명을 생략한다.

IV. 실험 결과

4.1 실험 환경

실험은 카페 가입을 위한 숫자로만 구성 된 캡챠로 진행하였다. 실험 결과, 1,000개의 이미지 중 368개의 캡챠에 대해 정확히 맞추어 36.8%의 공격 성공률을 보였다.

테스트를 위해 해당 사이트 캡챠[15] 총 1,000개의 이미지를 다운 받아 테스트하였으며, 성능 측정을

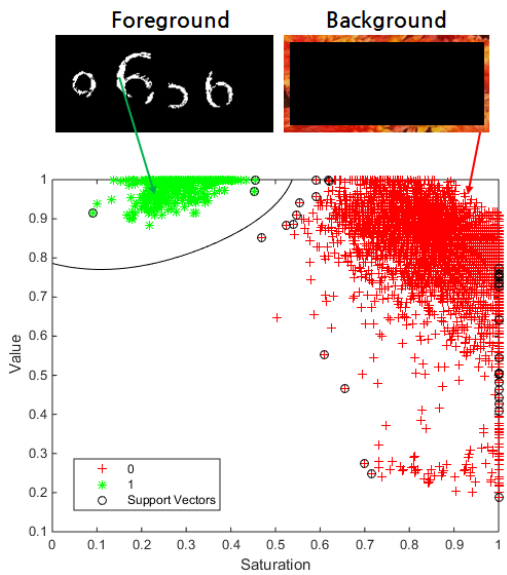


Fig. 12. SVM (before classification)

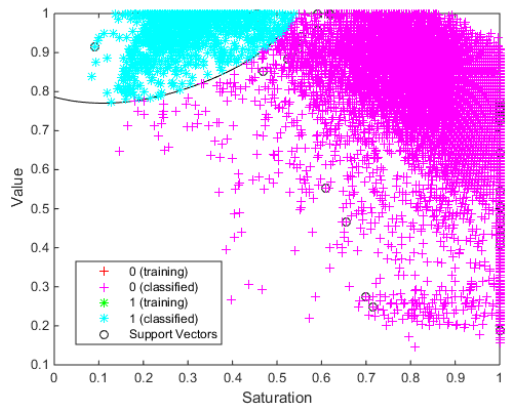


Fig. 13. SVM (after classification)

Fig. 14. Final Result

위해 캡챠 글자를 눈으로 직접 확인하여 실제 글자를 csv 파일에 기록하였다. 해당 이미지와 csv 파일은 연구실 홈페이지[16]에 업로드 하였다.

실험을 진행한 환경은 Table 1.과 같다. 실험은 MATLAB 2014b을 이용하여 진행하였으며, 분류 알고리즘인 SVM은 MATLAB 2014b 버전의 빌트

Table 1. Performance

Develop Environment	MATLAB 2014b
Spec	Intel(R) Core(TM) i7-3770 CPU @ 3.40GHz, 8GB
OS	Windows 7

인 함수로 구현하였다. 문자 인식 알고리즘인 CNN 구현을 위해 DeepLearnToolbox[17]를 사용하였으며, LBP 구현을 위해 [18]을 사용하였다.

4.2 실험 결과

실험은 4자리부터 8자리까지 각각 분류하여 실험한 것이 아닌 실제 랜덤하게 발생한 캡챠들을 이용해 실험하였고, 분류한 결과를 Table 2.에 나타내었다. 총 1,000개의 캡챠 중 368개의 캡챠에 대해 정확히 맞추어 36.8%의 공격 성공률을 보였으며, 공격 속도는 캡챠 한개당 평균 1초 내외로 3초에 1번꼴로 공격이 가능하다.

최근 기존 공격 방법[10]은 93%가 넘는 공격 성공률을 보여 본 논문에서 제시한 알고리즘의 성능과 차이가 많이 난다. 그러나 [10]에서 제시한 방법은 자연 배경이 없고 글자 수가 고정된 예전 버전의 캡챠에 대해서만 적용이 가능한 방법으로 새로운 캡챠에 대해 적용하면 글자 추출 자체가 안 되므로 공격이 불가능하다.

즉 [10]의 방법으로 새로운 캡챠에 대한 실험이 힘들며, 본 알고리즘의 공격 성공률과 비교하는 것은 무리가 있다.

Table 2. Accuracy rates

# of characters	Accuracy(%)	True / Total
4	62.2%	130 / 209
5	63.2%	127 / 201
6	33.7%	65 / 193
7	20%	40 / 200
8	3%	6 / 197
total	36.8%	368 / 1000

4.3 결과 분석 및 본 알고리즘의 한계점

Table 2. 결과를 보면 글자 수가 6개 이상으로 증가하면서 공격 성공률이 낮아지게 된다. 그 이유는 글자 중 하나라도 잘못 인식하게 될 경우 공격에 실패하기 때문이다.

Fig. 15.는 8자리 글자를 가진 캡챠에 대한 공격 실패 결과중 하나로 Fig. 15.(a) 캡챠는 실제 글자가 72112453 이지만 인식 결과는 7112413 이었고, Fig. 15.(b) 캡챠는 실제 글자가 93967152 이지만 인식 결과는 9317122 였다. 각각 6개, 5개의 글자를 맞추었지만 결과적으로는 공격에 실패하였다.

즉 8자리 중 7자리를 맞추어도 한 자리가 틀리면 공격에 실패하게 되므로 글자 수가 많아질수록 공격에 실패할 확률이 높아지게 된다. 또 한 가지로는 글자 수가 많아질수록 글자 크기가 작아지게 되어 글자 추출이 쉽지 않은데 글자 추출을 실패하는 원인은 크게 세 가지 경우로 볼 수 있다.

첫째, 실제 글자를 잡음으로 보고 제거하는 경우이다. Fig. 15.(a) 캡챠는 두 번째 자리인 2의 크기가 다른 글자에 비해 상대적으로 작게 형성되었고, 이를 잡음으로 인식하여 제거하였다. 즉 글자 수가 많아지면서 글자 크기는 줄어들게 되는데 작은 글씨까지 고려하여 잡음 제거 사이즈를 줄이게 되면 진짜 잡음까지 제거를 하지 못하는 경우가 발생하게 된다. 이밖에도 Fig. 15.(b) 캡챠의 일곱 번째 자리인 5는 윗부분과 아랫부분이 분리되어 추출되었는데 크기가 작은 윗부분을 잡음으로 인식하여 제거하였고 이에 5를 2로 잘못 인식하였다.

둘째, 잡음을 추출하였는데 제거되지 못하는 경우이다. Fig. 15.(a) 캡챠의 일곱 번째 자리인 5 밑에는 가로줄의 잡음이 발생하였는데, 해당 잡음이 5와 미세하게 붙어있기 때문에 잡음이 제거되지 못하였고 이에 붙어 있는 자체를 하나의 글자로 인식하게 되어 5를 1로 잘못 인식하였다.

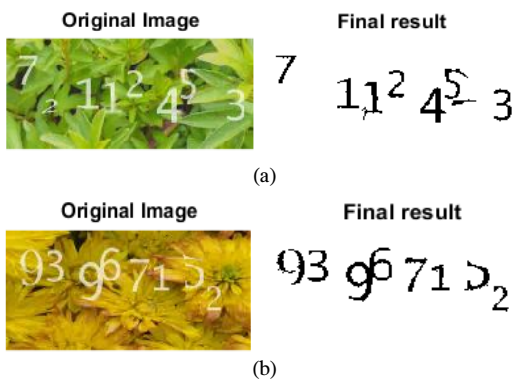


Fig. 15. Examples of poorly extracted characters

셋째, 글자와 글자가 붙어 있어 분리하지 못한 채 붙어있는 글자를 인식하게 되는 경우로 Fig. 15.(b) 캡챠는 세 번째 자리인 9와 네 번째 자리인 6이 붙어 있어 96 두 글자를 1로 잘못 인식하였다.

사실 해당 캡챠는 글자가 붙어 있는 경우는 많이 없고, 대부분은 실제 글자를 잡음으로 보고 제거하는 첫 번째 경우와 잡음이 제대로 제거되지 않아 잡음을 글자로 인식하는 두 번째 경우이다. 잡음 문제를 잘 해결해야만 해당 캡챠에 대한 효과적인 공격이 가능하며 이 점은 향후 연구 과제로 남겨둔다.

V. 결 론

본 논문에서는 자연 배경을 지닌 텍스트 기반 캡챠의 글자를 SVM과 CNN을 이용해 효과적으로 분리할 수 있음을 선보였다. 실험 결과 36.8% 확률로 공격이 가능하였으며, 이는 3번 접근시 1번은 해당 캡챠가 무력화 될 수 있음을 통계적으로 의미한다. 또한 실험은 카페 가입을 위한 숫자로만 이루어진 캡챠로 진행하였지만, 알파벳과 숫자로 구성된 다중 로그인 방지 캡챠 역시 똑같은 원리로 공격이 가능하기에 이 역시 안전하지 않음을 알 수 있다.

따라서 해당 사이트는 캡챠가 공격 당하지 않도록 빠른 시일 내에 현재 사용하고 있는 캡챠의 디자인을 변경해야 할 것으로 권고된다. 글자 분리가 힘든 복합적인 요소를 사용하여 캡챠의 난이도를 높이는 것이 필요하며 그렇다고 사람조차 알아보기 힘든 가독성이 떨어지는 캡챠를 만들게 되면 사용자의 불편을 초래할 수 있으므로 이 점 역시 고려해서 개선해야 한다.

References

[1] Von Ahn, Luis, et al. "CAPTCHA: Using hard AI problems for security," Advances in Cryptology—EUROCRYPT 2003. Springer Berlin Heidelberg, vol. 2656, pp. 294-311, May. 2003.

[2] Hernandez-Castro, Carlos Javier, and Arturo Ribagorda. "Pitfalls in CAPTCHA design and implementation: The Math CAPTCHA, a case study," computers & security, vol. 29, no. 1, pp. 141-157, Feb. 2010.

[3] Soupionis, Yannis, and Dimitris Gritzalis. "Audio CAPTCHA: Existing solutions assessment and a new implementation for VoIP telephony," Computers & Security, vol. 29, no. 5, pp. 603-618, Jul. 2010.

[4] Kalsoom, Sajida, Sheikh Ziauddin, and Abdul Rehman Abbasi. "An image-based CAPTCHA scheme exploiting human appearance characteristics," KSII Transactions on Internet and Information Systems (TIIS), vol. 6, no. 2, pp. 734-750, Feb. 2012.

[5] Bursztein, Elie, Matthieu Martin, and John Mitchell. "Text-based CAPTCHA strengths and weaknesses," Proceedings of the 18th ACM conference on Computer and communications security. ACM, pp. 125-138, Oct. 2011.

[6] Gunn, Steve R. "Support vector machines for classification and regression," ISIS technical report 14, May. 1998.

[7] Lee, Yuchun. "Handwritten digit recognition using k nearest-neighbor, radial-basis function, and backpropagation neural networks," Neural computation, vol. 3, no. 3, pp. 440-449, Mar. 1991.

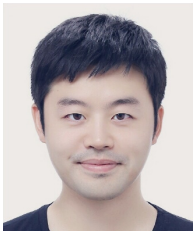
[8] Ciresan, Dan Claudiu, et al. "Convolutional neural network committees for handwritten character classification," Document Analysis and Recognition (ICDAR), 2011 International Conference on. IEEE, pp. 1135-1139, Sep. 2011.

[9] SungHo Kim, DaeHun Nyang and KyungHee Lee. "Breaking character-based CAPTCHA using color information," Journal of The Korea Institute of Information Security & Cryptology(JKIISC), 19(6), pp. 105-112, Dec. 2009.

[10] DaeHun Nyang, YongHeon Choi, SeokJun Hong and Kyunghee Lee, "Analysis of Naver CAPTCHA with

Effective Segmentation." Journal of The Korea Institute of Information Security & Cryptology(JKIISC), 23(5), pp. 909-917, Oct. 2013.
〔11〕 Smith, Alvy Ray. "Color gamut transform pairs," ACM Siggraph Computer Graphics. vol. 12, no. 3, pp. 12-19, Aug. 1978.
〔12〕 Ojala, Timo, Matti Pietikäinen, and David Harwood. "A comparative study of texture measures with classification based on featured distributions," Pattern recognition, vol. 29, no. 1, pp. 51-59, Jan. 1996.
〔13〕 LeCun, Yann, Koray Kavukcuoglu, and Clément Farabet. "Convolutional networks and applications in vision." Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on. IEEE, pp. 253-256, May. 2010.
〔14〕 Recognition demos using CNN, "LeNet-5, convolutional neural networks", http://yann.lecun.com/exdb/lenet/
〔15〕 Naver CAPTCHA link, https://nid. naver.com/login/image/captcha/nhncaptchav4.gif?key=??
〔16〕 1,000 CAPTCHAs datasets, http:// multimedia.korea.ac.kr/uploads/TEST_DATA_1000_image_set.zip
〔17〕 CNN Library for MATLAB, https:// github.com/rasmusbergpalm/DeepLearnToolbox
〔18〕 LBP Library for MATLAB, https:// github.com/adikhosla/feature-extraction/tree/master/features

〈저자소개〉



김 재 환 (Jaehwan Kim) 학생회원
2014년 2월: 한국외국어대학교 컴퓨터공학과 졸업
2014년 3월~현재: 고려대학교 정보보호학과 석사과정
〈관심분야〉 머신러닝, 패턴인식, 영상처리, 컴퓨터보안

김 수 아 (Suah Kim) 학생회원
2013년 2월: University of Waterloo, Combinatorics and Optimization
2013년 3월~현재: 고려대학교 정보보호학과 석박통합과정
〈관심분야〉 머신러닝, 컴퓨터보안, 영상처리

김 형 중 (Hyoung Joong Kim) 종신회원
1978년 2월: 서울대학교 전기공학 졸업
1986년 2월: 서울대학교 제어계측공학 석사 졸업
1989년 2월: 서울대학교 제어계측공학 박사 졸업
1989년~2006년: 강원대학교 교수
2006년~현재: 고려대학교 정보보호대학원 교수
〈관심분야〉 컴퓨터보안, 패턴인식, 가역정보은닉