

## ***Completely Automated Public Turing Test to Tell Computers and Humans Apart (CAPTCHA) Menggunakan Pendekatan Drag and Drop***

**Muhammad Fikry<sup>1</sup>, M. Iskandarsyah<sup>2</sup>**

<sup>1,2</sup> Jurusan Teknik Informatika, Fakultas Sains dan Teknologi, UIN Sultan Syarif Kasim Riau  
Jl. HR. Soebrantas No. 155 Simpang Baru, Panam, Pekanbaru, 28293  
Email: mfikry1980@yahoo.com, iskandarsyah1992@gmail.com

(Received: 5 Januari 2016; Revised: 12 Februari 2016; Accepted: 3 Februari 2016)

### **ABSTRAK**

CAPTCHA merupakan suatu tes untuk membedakan pengguna situs *web*, apakah manusia atau program komputer. Umumnya, CAPTCHA dijawab dengan cara mengetikkan jawaban. Permasalahannya adalah aktifitas pengetikan dapat dilakukan baik oleh manusia maupun oleh program komputer. CAPTCHA dengan menggunakan pendekatan *drag and drop* diharapkan dapat meningkatkan keamanan situs *web*. Hal ini dikarenakan untuk menembus situs *web*, program komputer harus melakukan aktifitas *drag and drop*, dan untuk saat ini, aktifitas *drag and drop* masih memerlukan keterlibatan manusia. Hasil penelitian membuktikan bahwa CAPTCHA dengan menggunakan pendekatan *drag and drop* mudah diselesaikan oleh manusia. Rata-rata nilai tingkat kenyamanan yang diberikan pengguna saat pengujian pertama dan kedua yaitu untuk teks sebesar 8,425 dengan rata-rata waktu penyelesaian 12,245 detik, gambar sebesar 7,050 dengan rata-rata waktu penyelesaian 27,420 detik dan video sebesar 7,825 dengan rata-rata waktu penyelesaian 15,495 detik.

**Kata Kunci:** CAPTCHA, *drag and drop*

### **ABSTRACT**

*CAPTCHA is a test to determine whether or not the user is human. Generally, CAPTCHA is answered by typing the answer. The problem is the activity of typing can be done either by human or by computer program. CAPTCHA with a drag and drop approach can improve web site security. This is because to penetrate the web site, computer program must perform drag and drop activity, and for now, it still required human involvement. This research proves that CAPTCHA using drag and drop approach easily solved by humans. The average value from the first and second test is 8,425 with mean completion time of 12,245 seconds for text-based CAPTCHA, 7,05 with mean completion time of 27,42 seconds for image-based CAPTCHA and 7,825 with mean completion time of 15,495 for video-based CAPTCHA.*

**Keywords:** CAPTCHA, *drag and drop*

---

#### **Corresponding Author**

Muhammad Fikry  
Program Studi Teknik Informatika, Fakultas Sains dan Teknologi,  
Universitas Islam Negeri Sultan Syarif Kasim Riau  
Email: mfikry1980@yahoo.com

---

### **Pendahuluan**

Menurut Kluever [2] yang dikutip oleh Setiawan [1], serangan otomatis terhadap *text-based* CAPTCHA telah berhasil dilakukan sebesar 20% terhadap Google CAPTCHA, 30-35% terhadap Microsoft CAPTCHA, 35% terhadap Yahoo! CAPTCHA. Sementara itu, serangan otomatis terhadap *audio-based* Google CAPTCHA telah berhasil dilakukan sebesar 90%. Untuk

mengatasi serangan-serangan tersebut, diperlukan sebuah varian baru atau sebuah pendekatan baru dari CAPTCHA agar penyerang tidak lagi dapat mengambil keuntungan yang akan merugikan pihak yang diserang.

Kluever [2] telah berhasil membuat dan menguji keamanan serta penggunaan dari varian baru CAPTCHA yaitu CAPTCHA berbasis video. Video-video yang digunakan berasal dari situs youtube.com. Kluever memanfaatkan *metadata* berupa tag yang ada pada

halaman video tersebut sebagai *challenge* dari CAPTCHA yang dibuatnya sehingga menyulitkan komputer untuk menjawabnya. Disisi keamanan, CAPTCHA berbasis video yang dibuat oleh Kluever dapat dikatakan lebih baik apabila dibandingkan dengan tipe-tipe CAPTCHA lainnya, berdasarkan rata-rata jumlah keberhasilan terhadap serangan yang dilakukan. Perbandingan rata - rata keberhasilan manusia dan mesin dalam menyelesaikan beberapa jenis CAPTCHA yaitu berbasis teks, gambar dan video dapat dilihat pada Tabel 1 berikut ini.

Tabel 1. Perbandingan rata-rata keberhasilan manusia dan mesin dalam menyelesaikan beberapa tipe CAPTCHA [2]

CAPTCHA	Type	Success Rates	
		Human	Machine
Microsoft	Text-based	0.90	0.60
Baffletext	Text-based	0.89	0.25
Handwritten	Text-based	0.76	0.13
ASIRRA	Image-based	0.99	0.10
Video	$\tau = (15, 0.003, T, T)$	0.77	0.02
	$\tau = (25, 0.006, T, T)$	0.86	0.05
	$\tau = (90, 0.006, T, T)$	0.90	0.13
		0.90	0.13

Canter [3], berasal dari universitas yang sama dengan Kluever, juga berusaha untuk memecahkan *challenge* dari CAPTCHA yang dibuat oleh Kluever. Dari 13 sampel *challenge* yang disediakan oleh Kluever, Canter dapat memecahkan 9 sampel *challenge* dengan menggunakan program komputer (umumnya dikenal sebagai *bot*) miliknya. Program *bot* tersebut dibuat oleh Canter dengan menggunakan Tesseract OCR engine. Sebelum menggunakan engine tersebut, Canter menggunakan program yang bisa melakukan *snapshot* pada video, dengan tujuan memperoleh potongan-potongan video dalam bentuk gambar. Selanjutnya, gambar-gambar tersebut diproses oleh program yang dapat mengenali dan menjawab *challenge* CAPTCHA dari Kluever tersebut.

Selain Canter, Elie Bursztein, seorang peneliti dari Universitas Stanford juga melakukan pemecahan *challenge* dari CAPTCHA. CAPTCHA yang menjadi targetnya yaitu NuCAPTCHA. Hasilnya 90% dari *challenge* NuCAPTCHA dapat ditembus oleh program *bot* miliknya. Program *bot* tersebut diberi nama DeCAPTCHA [3].

Selanjutnya, Desai dan Patadia [4] merekomendasikan suatu pendekatan baru untuk membuat dan meningkatkan keamanan pada *text-based* CAPTCHA yaitu menggunakan pendekatan *drag and drop*. Pendekatan tersebut memanfaatkan aksi *mouse* untuk membedakan antara manusia dan komputer *bot*. Dengan menggunakan pendekatan ini, pengguna tidak lagi mengetikkan jawaban dari *challenge* CAPTCHA pada *textbox*, melainkan menyeret (*drag*) dan meletakkan (*drop*) jawaban di tempat yang telah disediakan. Dengan

menggunakan pendekatan *drag and drop* tersebut, CAPTCHA yang dihasilkan akan lebih sederhana dalam pengoperasiannya, lebih mudah bagi manusia untuk menyelesaikannya, lebih rendah penggunaan *bandwidth*-nya dan mudah untuk diterapkan serta di-*maintenance* dan tentunya akan meningkatkan keamanan CAPTCHA. Peningkatan keamanan dimungkinkan karena aktifitas *drag and drop* secara otomatis masih sulit dilakukan oleh *bot*, dimana aktifitas *drag and drop* masih memerlukan intervensi manusia. Namun demikian, pendekatan yang direkomendasikan oleh Desai dan Patadia tersebut belum menjabarkan secara jelas tingkat kenyamanan pengguna dalam menyelesaikan CAPTCHA yang menggunakan pendekatan ini.

Berdasarkan uraian-uraian diatas, dalam penelitian ini akan dilakukan pembuatan CAPTCHA (teks, gambar dan video) dengan menggunakan pendekatan *drag and drop* yang nantinya akan menambah varian dari CAPTCHA yang sudah ada dan juga akan dilakukan pengujian untuk mengetahui seberapa besar tingkat kenyamanan pengguna dalam menyelesaikan CAPTCHA, baik teks, gambar maupun video.

### CAPTCHA

Menurut Setiawan [1], CAPTCHA merupakan suatu teknik untuk membedakan antara manusia dengan komputer *bot* di *internet*. Biasanya, CAPTCHA dipergunakan pada saat manusia melakukan proses registrasi *account* (misalnya pendaftaran sebagai anggota pada suatu situs *web*) atau pada saat manusia melakukan pengisian data (misalnya komentar pada berita artikel, atau konten pada buku tamu) untuk mencegah komputer *bot* menciptakan atau mengisi data yang tidak valid.

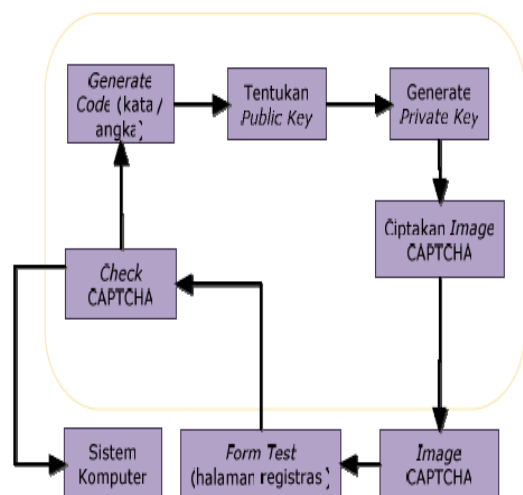
Menurut Kluever [2], A CAPTCHA is a variation of the Turing test, in which a challenge is used to distinguish humans from computers ('bots') on the internet". (CAPTCHA adalah salah satu variasi dari tes Turing [7], dimana suatu tantangan digunakan untuk membedakan manusia atau komputer ('bots') di *internet*).

CAPTCHA pada dasarnya merupakan sebuah program yang sebagian besar manusia mampu untuk melewatinya, akan tetapi komputer tidak dapat melewatinya [5]. CAPTCHA banyak digunakan oleh penyedia *web mail* seperti Yahoo!, dan Gmail. CAPTCHA dikembangkan untuk mencegah *bot* yang menciptakan ratusan *email account* untuk mengirimkannya ke pengguna [6]. Bot ini digunakan oleh *spammer* untuk melakukan penyerangan terhadap sistem dengan menggunakan HTTP POST *request submission*. Program robot akan mengambil nilai variabel yang terdapat pada HTTP POST *request* tersebut dari *form* yang akan di-*submit* sebelumnya dan mengirimkannya kembali secara berulang-ulang. Penyerang dapat dengan mudah melakukan hal tersebut dengan menulis *script* menggunakan bahasa Perl.

Berdasarkan rekomendasi dari Carnegie Mellon University, Palo Alto Research Center, Microsoft Research, karakteristik dari penggunaan CAPTCHA harus bersifat [5]:

1. **Automated.** Tantangan yang dilakukan harus dihasilkan secara otomatis dan dapat ditingkatkan kesulitannya dengan mudah oleh komputer.
2. **Open.** Algoritma dari tantangan yang dilakukan harus bersifat publik.
3. **Usable.** Tantangan harus mudah untuk diselesaikan oleh manusia dalam waktu yang wajar.
4. **Secure.** Tantangan yang dilakukan harus sulit bagi komputer untuk memecahkan algoritmanya.

CAPTCHA digunakan sebagai alternatif Reverse Turing Test untuk memproteksi suatu situs web. Gambar 1 berikut ini menggambarkan cara kerja CAPTCHA secara umum.



Gambar 1. Metodologi penelitian

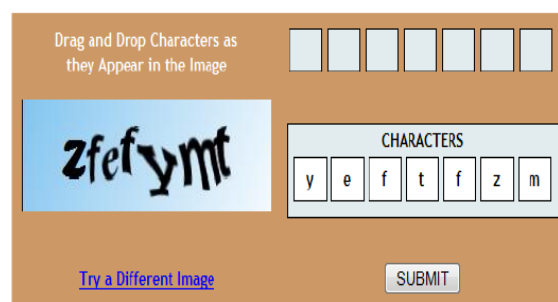
Berikut penjelasan gambar diatas:

1. Rangkaian kode di-generate berupa huruf dan angka atau sebuah kata, kemudian suatu *image-based* CAPTCHA di-generate berdasarkan rangkaian kode tersebut. Sesuai dengan algoritma *public key cryptography* akan di-generate *private key* yang hanya dapat berpasangan dengan rangkaian kode yang dihasilkan tersebut (*public key*).
2. *Private key* akan di-render sebagai *hidden field* pada form, dan harus ada *input box* yang akan menerima masukan dari pengguna berupa *public key* atau *image CAPTCHA* yang ditampilkan.
3. *Private key* dan *public key* yang akan di-submit oleh pengguna kemudian dibandingkan oleh sistem dengan algoritma *public key cryptography*. Jika ada kecocokan, berarti CAPTCHA sah dan jika tidak ada kecocokan,

berarti ada indikasi yang melakukan *submit* bukan manusia.

### Pendekatan Drag and Drop

Pendekatan *drag and drop* adalah suatu pendekatan pembuatan CAPTCHA yang direkomendasikan oleh Arpan Desai dan Pragnesh Patadia yang menggunakan tindakan *mouse* untuk membedakan antara manusia dan komputer [4]. Tindakan *mouse* yang digunakan yaitu aktivitas klik objek bergerak yang bersifat spesifik atau klik pada bagian tertentu dari sebuah gambar untuk berhasil lulus dari tes CAPTCHA. Dalam tes CAPTCHA *drag and drop* ini, pengguna harus menyelesaikan gambar CAPTCHA normal, tetapi pengguna tidak bisa mengetik jawaban ke kotak teks (*textbox*) melainkan harus menyeret (*drag*) dan meletakkan (*drop*) ke blok kosong masing-masing blok seperti yang ditampilkan pada gambar 2.



Gambar 2. DnD CAPTCHA *draggable characters block and dropbox* [4]

Adapun tujuan utama penggunaan CAPTCHA *drag and drop* menurut Desai dan Patadia [4] yaitu:

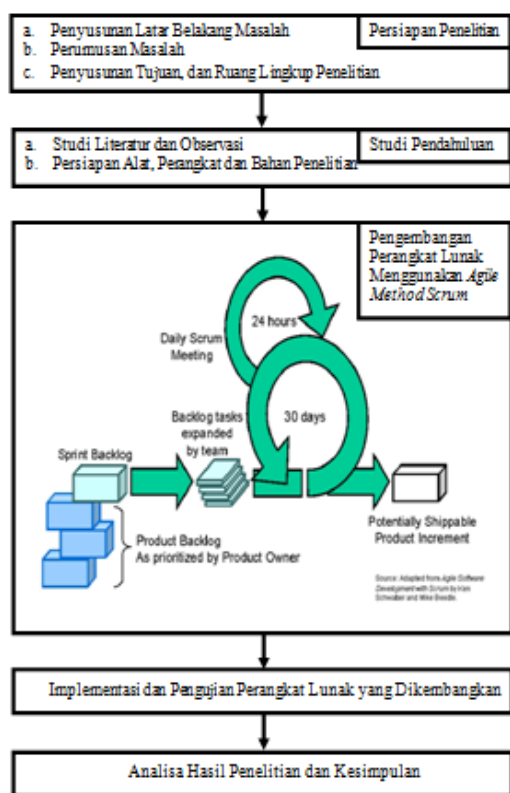
1. **Kesederhanaan operasi**  
 Pengguna tidak harus memiliki keterampilan tambahan untuk menjawab tes, sehingga orang awam pun mampu untuk menjawab *challenge* tersebut.
2. **Uji harus mudah bagi manusia**  
 Untuk *drag and drop* objek apapun sangat mudah bagi setiap manusia dan tidak membutuhkan kemampuan analisis atau teknis khusus.
3. **Uji harus sulit untuk komputer**  
 Untuk algoritma yang mengidentifikasi blok karakter dan untuk melakukan *drag and drop* secara otomatis saat ini, masih sulit dipecahkan oleh *bot* sebab untuk melakukan *drag and drop* masih memerlukan intervensi dari manusia.
4. **Keamanan tinggi dengan bandwidth yang lebih rendah.**  
 Dengan menggunakan pendekatan *drag and drop*, maka serangan akan dapat diminimalisir dan kebutuhan akan *bandwidth* akan dapat dikurangi sebab tidak memerlukan latar belakang gambar yang mencolok sehingga

*bandwidth* yang digunakan pun akan jadi lebih efisien dibandingkan menggunakan teknik animasi. Saat ini, pendekatan *drag and drop* menggunakan kombinasi dasar HTML dan JavaScript, sehingga tidak akan mengkonsumsi *bandwidth* yang tinggi.

5. **Mudah untuk diterapkan dan di-maintaince**  
 Saat ini, perancangan menggunakan HTML, JavaScript dan sedikit bantuan CSS yang mana mudah untuk diimplementasikan.

### Metode Penelitian

Tahapan-tahapan yang dilakukan dalam penelitian ini diperlihatkan pada Gambar 3.



Gambar 3. Metode penelitian

### Hasil dan Pembahasan

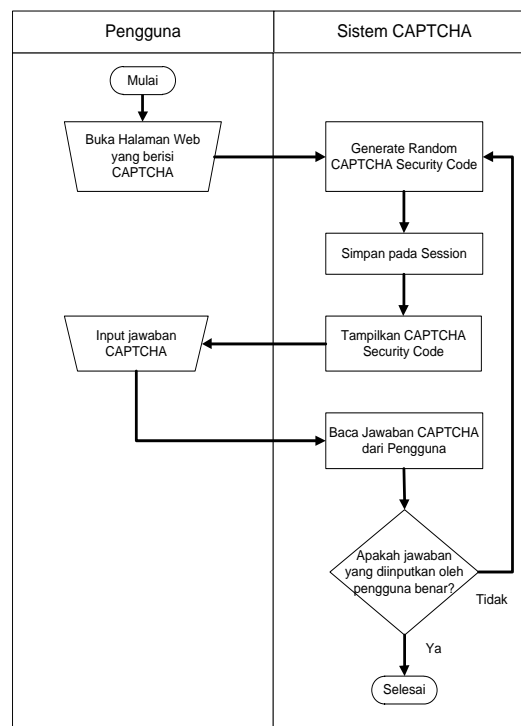
#### Spesifikasi Perangkat Lunak

Setelah melakukan analisa terhadap CAPTCHA konvensional, dimana pengguna diminta untuk mengetikkan jawaban, dapat disimpulkan beberapa hal sebagai berikut.

1. CAPTCHA terdiri atas tiga bagian yaitu :
  - a. Bagian generator (pembangkit) *challenge* CAPTCHA
  - b. Bagian penampil *challenge* CAPTCHA
  - c. Bagian pengecek jawaban dari *challenge* CAPTCHA .

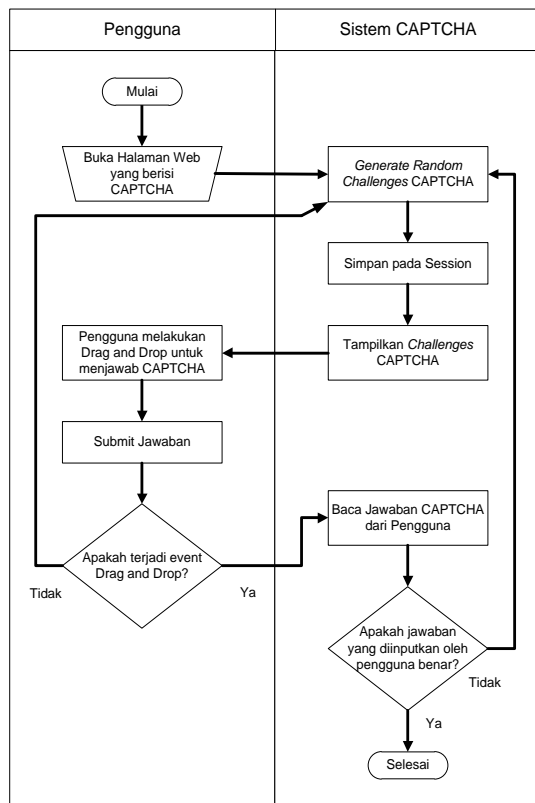
2. CAPTCHA memiliki empat proses utama yaitu :
  - a. *Generate challenge* acak
  - b. Penyimpanan *challenge* pada *session*
  - c. Penampila *challenge* pada *web browser*
  - d. Pengecekan jawaban terhadap *challenge* CAPTCHA.
3. Bagian penampil *challenge* CAPTCHA ditempatkan pada halaman *web* yang menampilkan *challenge* yang dihasilkan oleh *generator*.
4. Pengguna mengisi jawaban dari *challenge* CAPTCHA yang ditampilkan dan mengirimkannya ke sistem CAPTCHA,
5. Sistem CAPTCHA membaca jawaban yang dimasukkan oleh pengguna serta melakukan perbandingan terhadap solusi sebenarnya.
6. Jika jawaban yang diberikan oleh pengguna sama dengan solusi sebenarnya maka pengguna dinyatakan valid dan proses yang diinginkan oleh pengguna dapat dilanjutkan. Akan tetapi, jika solusi jawaban yang diberikan oleh pengguna tidak sama dengan solusi sebenarnya maka pengguna dinyatakan tidak valid, dan suatu notifikasi akan dimunculkan kepada pengguna. Biasanya pengguna akan disajikan *challenge* yang baru.

Gambar 4 memperlihatkan *flowchart* untuk sistem CAPTCHA konvensional.



Gambar 4. Flowchart untuk Sistem CAPTCHA Konvensional

Gambar 5 memperlihatkan *flowchart* untuk sistem CAPTCHA *drag and drop*.



Gambar 5. Flowchart untuk Sistem CAPTCHA Drag and Drop

Hampir sama dengan sistem CAPTCHA konvensional, sistem CAPTCHA *drag and drop* masih menggunakan menggunakan *generator random* untuk memperoleh *challenge* CAPTCHA. Perbedaannya terletak pada cara pengguna menjawabnya yaitu pengguna tidak lagi menuliskan jawaban dari *challenge* CAPTCHA di *textbox*, melainkan pengguna memilih jawaban yang tepat dengan melakukan aktifitas *drag and drop*.

Aktifitas *drag and drop* yang dimaksud berupa penggunaan perangkat penunjuk (misalnya *mouse*, *touchpad* dan *trackpad*) untuk memilih suatu obyek pada layar dengan cara meraih obyek tersebut dan menyeretnya ke lokasi yang berbeda atau ke obyek lain. Setelah pengguna melakukan aktifitas *drag and drop*, sistem CAPTCHA mencocokkan jawaban pengguna dengan solusi sebenarnya, jika jawaban cocok dan telah terjadi aktifitas *drag and drop* maka sistem CAPTCHA akan menyimpulkan bahwa yang melakukan inputan adalah manusia, dan sebaliknya jika tidak ada maka yang melakukan inputan adalah komputer *bot*.

Pada *video-based* CAPTCHA, pengacakan dengan *generator random* berupa pengacakan video beserta pertanyaannya sehingga akan didapatkan sebuah *challenge* CAPTCHA video yang acak dengan pertanyaan yang juga teracak. Untuk menjawab *challenge* tersebut aktifitas *drag and drop* akan sangat dibutuhkan. Jumlah video yang digunakan menjadi konten CAPTCHA pada penelitian ini berjumlah 5 video

dengan 5 pertanyaan yang berbeda sehingga setelah dilakukan pengacakan menggunakan algoritma *generate random* akan menghasilkan 25 *challenge* CAPTCHA yang berbeda.

### Hasil Implementasi

Gambar 6 memperlihatkan salah satu *challenge* untuk *text-based* CAPTCHA dengan *drag and drop*.



Gambar 6. Implementasi CAPTCHA Teks Drag and Drop

Pada *text-based* CAPTCHA, terdapat gambar hasil dari *generator challenge* CAPTCHA dan blok-blok karakter yang ditampilkan secara acak. Jika pengguna ingin menyelesaikan CAPTCHA, maka pengguna harus menyusun secara berurutan blok-blok karakter tersebut dengan melakukan aktifitas *drag and drop* sampai blok-blok karakter tersusun dengan benar sesuai dengan susunan yang terdapat pada gambar hasil dari *generator challenge* CAPTCHA.

Gambar 7 memperlihatkan salah satu *challenge* untuk *image-based* CAPTCHA berupa *puzzle* dengan *drag and drop*.

### Drag and Drop Gambar Untuk Menyelesaikan CAPTCHA Puzzle Berikut !!!



Gambar 7. Implementasi CAPTCHA Gambar Puzzle Drag and Drop

Pada *image-based* CAPTCHA berupa *puzzle*, sistem CAPTCHA menggunakan *generator challenge* CAPTCHA dengan cara memotong gambar, kemudian melakukan pengacakan terhadap gambar yang telah dipotong dan disajikan pada penampil CAPTCHA agar diselesaikan oleh pengguna. Untuk menyelesaikan



CAPTCHA ini, pengguna menyusun gambar tersebut dengan melakukan aktifitas *drag and drops* sampai terbentuk gambar yang utuh.

Gambar 8 memperlihatkan *video-based* CAPTCHA yang terdiri atas beberapa komponen, mulai dari atas sampai ke bawah, yaitu pemutar video, pertanyaan yang menjadi *challenge* video, *drop box* dan sejumlah alternatif pilihan jawaban.



Gambar 8. Implementasi CAPTCHA Video *Drag and Drop*

Pada *video-based* CAPTCHA, sistem CAPTCHA menampilkan video beserta pertanyaan yang menjadi *challenge* CAPTCHA. Pertanyaan tersebut didasarkan pada informasi yang akan didapatkan oleh pengguna saat menonton cuplikan video yang ada. Setelah pengguna mengetahui jawaban dari *challenge* CAPTCHA tersebut, pengguna melakukan aktifitas *drag* terhadap salah satu gambar yang menjadi pilihan jawaban, kemudian melakukan aktifitas *drop* terhadap gambar tersebut ke *drop box*. Sistem CAPTCHA akan melakukan pengecekan apakah terjadi *event drag and drop*, dan apakah jawaban yang dipilih oleh pengguna benar.

### Hasil Pengujian

Pengujian dilakukan dengan menggunakan metode *blackbox*, dan User Acceptance Test (UAT) terhadap 10 orang sebanyak 3 kali untuk ketiga jenis CAPTCHA. Pengujian UAT dimaksudkan untuk mengetahui seberapa besar tingkat kenyamanan pengguna dalam menyelesaikan CAPTCHA teks, gambar dan video menggunakan *drag and drop* yang dibuat dan waktu penyelesaiannya. Skala nilai kemudahan yang digunakan yaitu nilai 1 - 4 dikatakan sulit, nilai 5 - 7 dikatakan mudah dan nilai 8 - 10 sangat mudah.

Dari pengujian- pengujian yang telah dilakukan dapat ditarik kesimpulan sebagai berikut:

1. Berdasarkan hasil pengujian dengan metode *blackbox*, dapat disimpulkan bahwa CAPTCHA yang dibuat berhasil berjalan sesuai dengan yang diharapkan dan adanya kesesuaian fungsi-fungsi yang telah diimplementasikan terhadap hasil analisa dan perancangan.
2. Berdasarkan hasil pengujian UAT (User Acceptance Test) yang telah dilakukan, dapat disimpulkan tingkat kenyamanan pengguna dalam menyelesaikan CAPTCHA *drag and drop* yang dibuat, sebagaimana diperlihatkan pada Tabel 2.

Tabel 2. Hasil Pengujian UAT

JENIS CAPTCHA DRAG AND DROP	TINGKAT KENYAMANAN			RATA-RATA WAKTU MENYELESAIKAN CAPTCHA (Detik)		
	UAT I	UAT II	Rata-rata	UAT I	UAT II	Rata-Rata
TEKS	8,3	8,55	8,425	11,87	12,62	12,245
GAMBAR	6,6	7,5	7,05	29,19	25,65	27,42
VIDEO	7,5	8,15	7,825	16,63	14,36	15,495

3. Berdasarkan pengujian keamanan yang dilakukan pada CAPTCHA teks *drag and drop* yang dilakukan, dapat ditarik kesimpulan bahwa CAPTCHA teks *drag and drop* yang dibuat tahan terhadap serangan berjenis *OCR attacks*.

### Kesimpulan

Berdasarkan penelitian yang telah dilakukan, dapat ditarik kesimpulan sebagai berikut :

1. Pengembangan CAPTCHA dengan menggunakan pendekatan *drag and drop* yang dilakukan telah menghasilkan sebuah varian baru CAPTCHA yang mudah diselesaikan oleh manusia.
2. Tingkat kenyamanan pengguna dalam menyelesaikan CAPTCHA dengan menggunakan pendekatan *drag and drop* dapat diketahui berdasarkan rata-rata nilai kemudahan yang diberikan pengguna dan rata-rata waktu penyelesaian saat pengujian UAT pertama dan kedua yaitu untuk *text-based* CAPTCHA sebesar 8,425 dengan rata-rata penyelesaian 12,245 detik, *image-based* CAPTCHA sebesar 7,050 dengan rata-rata penyelesaian 27,420 detik dan *video-based* CAPTCHA sebesar 7,825 dengan rata-rata penyelesaian 15,495 detik. Hal tersebut menandakan bahwa semua jenis CAPTCHA yang dibuat mudah untuk diselesaikan oleh manusia.

Untuk penelitian lanjutan, disarankan untuk meneliti lebih lanjut sisi keamanan dari CAPTCHA berbasis teks, gambar dan video tersebut, dengan membuat sebuah program *bot* yang tidak hanya dapat memecahkan *challenges* dari CAPTCHA *drag and drop* tetapi juga dapat melakukan aktifitas *drag and drop*.

#### Daftar Pustaka

- [1] Setiawan, E.B., Optimalisasi Keamanan Website Menggunakan CAPTCHA – Ad Video, Bandung. 2012.
- [2] Kluever, K.A., Balancing Usability and Security in a Video CAPTCHA, Rochester Institute of Technology, New York, 2008.
- [3] Canter, A., Assessing Threat Posed to Video CAPTCHA by OCR-Based Attacks, Rochester Institute of Technology, New York, 2013.
- [4] Desai, A., dan Patadia, P., Drag and Drop: A Better Approach to CAPTCHA, Gujarat, India, 2009.
- [5] Ahn, L.V., Blum, M., Hopper, N.J., dan Langford, J., CAPTCHA: Telling Humans and Computers Apart, *Advances in Cryptology, Eurocrypt '03*, volume 2656 of Lecture Notes in Computer Science, 294–311, 2003.
- [6] Mori, G. dan Malik, J., Recognizing objects in adversarial clutter— Breaking a visual CAPTCHA, *Proceedings of the Conference on Computer Vision and Pattern Recognition*, Juni 2003.
- [7] Turing, A.M., Computing Machinery and Intelligence, *Mind* 59, 236, 433–460, 1950.