

# Rancang Bangun *Game* Aksi dengan Integrasi dan Pengenalan Gambar Menggunakan Algoritma Ekstraksi Fitur SURF dan Klasifikasi SVM pada Perangkat Android

Bryan Nandriawan, Imam Kuswardayan, Nanik Suciati

Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember (ITS)

Jl. Arief Rahman Hakim, Surabaya 60111 Indonesia

*e-mail:* imam@its-sby.edu

**Abstrak**—Saat ini perangkat *mobile* berbasis Android dan iOS sangat digemari oleh anak-anak. Mereka yang telah memiliki perangkat *mobile* akan menghabiskan seluruh waktunya untuk menatap layar perangkat saat bermain. Akibat yang ditimbulkan adalah penurunan kesehatan mata dan kurangnya bersosialisasi dengan teman sebaya. Menggambar adalah hal yang disukai anak-anak. Dengan menggabungkan menggambar dengan permainan perangkat *mobile*, anak-anak akan tetap merasa bermain meski mereka sedang tidak menatap layar untuk menggambar. Tugas akhir ini bertujuan untuk membangun sebuah permainan yang dapat mengenali gambar dan menggunakan gambar tersebut dalam permainan. Metode pengenalan gambar dalam tugas akhir ini dilakukan dengan menggunakan metode *Edge Detection* dengan algoritma ekstraksi fitur SURF (*Speeded-Up Robust Features*) dan klasifikasi SVM (*Support Vector Machine*). Jenis permainan yang dibangun merupakan gabungan dari aksi dan *tower defense*. Hasil dari tugas akhir ini dibagi menjadi 3 yaitu hasil fungsionalitas permainan, hasil daya tarik permainan dan hasil akurasi pengenalan gambar. Semua fungsi yang dibuat pada permainan dapat berjalan dengan baik, dan permainan juga telah memiliki daya tarik yang memikat. Namun hasil akurasi pengenalan gambar masih kurang dari harapan penulis.

**Kata Kunci**— Android, Klasifikasi Gambar, Permainan, SURF, SVM

## I. PENDAHULUAN

PERANGKAT *mobile* berbasis Android dan iOS sangat digemari di masyarakat. Kemajuan teknologi yang sangat cepat dalam bidang ini, membuat banyak kebiasaan dan budaya masyarakat berubah. Seperti budaya anak-anak yang biasa bermain di luar rumah bersama teman, kini telah digantikan dengan bermain *gadget* sendirian. Banyaknya aplikasi, permainan dan hiburan yang ditawarkan dalam sebuah *gadget*, seringkali membuat pengguna menatap layar secara berlebihan. Perilaku ini seringkali ditiru oleh anak-anak yang juga tertarik pada perangkat pintar ini. Anak-anak generasi muda saat ini sudah sangat mengenal perangkat ini, bahkan tidak sedikit yang lebih mahir menggunakannya dibanding orang tua mereka. Namun anak-anak tidak bisa mengatur waktu mereka ketika mereka bermain. Mereka akan menghabiskan seluruh waktunya untuk menatap layar perangkat saat bermain, ketika tidak ada pengawasan dari orang tua. Hal ini tentu berbahaya bagi

kehatan mata dan kebiasaan bersosialisasi mereka. Karena itu, seharusnya setiap permainan memiliki sistem atau fitur untuk menghentikan pemain dalam bermain terlalu lama.

Untuk mengatasi persoalan tersebut, diperlukan permainan yang dapat menghentikan pemain secara langsung ataupun secara tidak langsung dalam menatap layar, namun pemain akan tetap merasa bermain dalam permainan tersebut. Dengan menugaskan pemain untuk menggambar suatu benda terlebih dahulu untuk dapat digunakan dalam permainan, akan membuatnya secara tidak langsung berhenti menatap layar perangkat. Pemain akan menggambar di kertas sesuai dengan objek yang diminta lalu gambar tersebut difoto dan dikenali. Gambar yang telah dikenali selanjutnya diubah menjadi objek di dalam permainan dan dapat digunakan. Tujuan dari pengenalan gambar adalah mencegah pemain berbuat curang dengan menggambar sesuatu yang tidak ditugaskan.

Dalam tugas akhir ini, pengenalan gambar menggunakan metode *Edge Detection* dengan algoritma ekstraksi fitur SURF (*Speeded-Up Robust Features*), *Bag of Words* dan klasifikasi SVM (*Support Vector Machine*) yang diharapkan dapat mengenali berbagai variasi (skala, rotasi dan bentuk) suatu jenis objek, sehingga tidak membatasi kreatifitas pemain. Pembentukan training model dengan ekstraksi fitur menggunakan algoritma SURF, *Bag of Words* dan klasifikasi menggunakan SVM pada beberapa jenis objek yang telah dipilih akan dilakukan terlebih dahulu. Lalu model akan dibandingkan dengan fitur dari gambar yang baru dibuat oleh pemain dengan mencari jarak terdekat. *Level* yang dibuat bertingkat dalam permainan diharapkan dapat mengasah kemampuan pemain dalam menggambar. Keseluruhan permainan ini tentunya akan membuat pemain merasa tetap bermain saat mereka beristirahat dari menatap layar perangkat.

Tujuan dari pembuatan tugas akhir ini adalah membuat permainan yang dapat membuat pemain berhenti menatap layar untuk menggambar sebuah benda yang kemudian difoto dan dikenali sebagai benda yang diminta.

## II. TINJAUAN PUSTAKA

### A. Game Aksi dan Tower Defense

*Game aksi* adalah salah satu *Genre* permainan yang seringkali membutuhkan refleks, kecepatan tangan, *timing*, dan koordinasi mata dengan tangan. Sedangkan *Tower Defense* adalah genre permainan yang memaksa pemain untuk bertahan dari serangan musuh yang berdatangan. Beberapa *game mobile* tidak hanya menyenangkan, namun juga edukatif. Sisi edukatif ini dikembangkan untuk membuat pemain menggunakan waktu bermainnya untuk sesuatu yang bermanfaat juga [1].

### B. OpenCV untuk Android

*OpenCV (Open Source Computer Vision)* adalah sebuah pustaka perangkat lunak yang ditujukan untuk pengolahan citra dinamis. Pustaka *OpenCV* mendukung penggunaan pada *Android Development*. Dengan menggunakan fungsi-fungsi yang ada dalam *OpenCV*, akan memudahkan pembuatan aplikasi yang membutuhkan visi komputer. Digunakan *asset Unity* dengan nama "*OpenCv for Unity*" untuk *thresholding* gambar dari kamera dan *resize* untuk memperkecil ukuran gambar.

### C. SURF, Bag of Words dan Klasifikasi SVM

*SURF: "Speeded up Robust Features"* adalah salah satu algoritma pencarian ujung ("*Edge Detection*"). *SURF* adalah versi lebih cepat dari *SIFT (Scale-Invariant Feature Transform)*. *SURF* dipilih karena sifatnya yang *robust* dalam mendeteksi fitur, dan karena tidak dibutuhkan perhitungan secara *real-time* dalam kasus ini. *SURF* juga dipilih karena memiliki *descriptor* dengan 64 nilai poin ambang yang lebih ringkas namun memiliki semua informasi penting, daripada *SIFT* yang memiliki *descriptor* dengan 128 nilai poin ambang. Pada *MATLAB* algoritma *SURF* yang digunakan telah ada dalam kelas *Bag of Words* [2].

*Bag of Words (BoW)* dalam visi komputer adalah penyerderhanaan dalam representasi fitur sebuah gambar dalam kata-kata. Pembentukan *BoW* akan melewati 3 tahap yaitu deteksi fitur, deskripsi fitur dan pembuatan *codebook*. Setelah dilakukan deteksi fitur dan deskripsi fitur, akan dihasilkan *descriptors* berupa *vector*. Terakhir, dengan melakukan *K-Means Clustering* pada semua *vector*, *vector* akan diubah ke bentuk *codewords* dalam *codebooks*. *Codebooks* berupa kumpulan *histogram vector* [3].

*Support Vector Machine (SVM)* adalah suatu teknik untuk melakukan prediksi, baik dalam kasus klasifikasi maupun regresi. *SVM* berada dalam satu kelas dengan *Artificial Neural Network (ANN)* dalam hal fungsi dan kondisi permasalahan yang bisa diselesaikan. Keduanya masuk dalam kelas *supervised learning*. Pemilihan *SVM* dalam kasus ini karena *SVM* efektif dalam *multidimensional spaces* [4].

### D. MATLAB Compiler SDK

*MATLAB compiler SDK* adalah salah satu fitur dari *MATLAB* dimana *sourcecode* *MATLAB* yang telah dibuat dapat diubah menjadi *C/C++ shared libraries*, *Microsoft .NET assemblies*, *Java class*, dan *Python packages*. Kemudian komponen baru tersebut dapat diintegrasikan dengan aplikasi yang dapat di *deploy* di *desktop*, *web*, dan sistem *enterprise*.

### E. Game Design

*Game Design* merupakan unsur-unsur yang ada dalam pembuatan sebuah permainan. Perancangan *game design* yang baik akan menghasilkan *game* yang baik pula. *Game design* dapat dibagi menjadi 2 yaitu *Game Mechanics* dan *Game Dynamics*. *Game Mechanics* adalah suatu desain metode atau aturan yang dibuat untuk pemain berinteraksi. Metode dan aturan ini dibuat dengan tujuan agar pemain merasa cukup tertantang dan terhibur dalam memainkan suatu *game*. *Game Dynamics* atau yang sering disebut *gameplay* adalah tujuan yang diberikan kepada pemain untuk dapat menyelesaikan *game* atau *level* yang ada [5].

## III. ANALISIS DAN PERANCANGAN

### A. Analisis Permainan

Permainan ini dibangun dengan tujuan untuk membantu pemain mengistirahatkan matanya dari layar, namun tetap memberikan kondisi berada dalam permainan. Permainan ini mengharuskan pemain menggambar objek yang dibutuhkan dalam permainan dengan gambar tangan pada kertas. Gambar kemudian difoto untuk kemudian dikenali dengan tujuan apakah objek tersebut sesuai dengan yang sedang diminta. Aplikasi ini dibangun untuk perangkat *mobile* dengan aplikasi *Unity*. Pengenalan gambar dilakukan terpisah dengan menggunakan halaman *web* lokal dan mengirimkan gambar untuk dikenali. Halaman *web* lokal dibuat dengan *ASP .Net* pada *Microsoft Visual Studio* dan telah tertanam *.Net assembly* dari *MATLAB* sebagai dasar algoritma pengenalan.

### B. Perancangan Game Mechanics Game Dynamics dan Peraturan Permainan

*Game mechanics* yang ada dalam permainan ini adalah:

1. Sistem pergantian peralatan. Sebelum memulai permainan, pemain dapat memilih peralatan yang pemain miliki untuk dibawa, sehingga memungkinkan pemain berganti senjata di dalam pertandingan.
  2. Sistem pembatas serangan. Pemain dalam pertandingan akan menggunakan peralatan untuk dapat menyerang musuh, namun setiap serangan yang dilakukan akan berharga sejumlah mana atau stamina. Mana dan stamina memiliki jumlah maksimum dan minimum dan pengurangannya tergantung dari jenis senjata yang digunakan. Jika mana atau stamina habis, maka pemain harus mengisinya dengan menekan tombol untuk mengisi mana atau stamina.
  3. Sistem *multi-touch*. Pemain dapat mengisi mana atau stamina bersamaan dengan menyerang musuh.
  4. Tingkat kesulitan bertahap. Tingkat kesulitan tiap *level* tidaklah *random* namun meningkat bertahap. Setiap pemain akan mendapatkan musuh yang sama pada *level* yang sama.
  5. Sistem *RPG (Role Playing Games)*. Sistem ini lebih spesifik kepada pemain yang dapat mendapatkan *gold* atau mata uang yang ada dalam permainan. Pemain mendapatkan *gold* setiap *level* terselesaikan. *Gold* dapat digunakan untuk meningkatkan kekuatan karakter pemain dan peralatannya.
- Game dynamics* pada permainan ini adalah:
1. Mengalahkan semua musuh yang datang, sebelum jumlah *HP (Hit Points)* pemain mencapai 0.

2. Mendapatkan *gold* untuk mempermudah permainan di *level* berikutnya.
3. Membuka peralatan baru dengan melakukan pengenalan gambar.

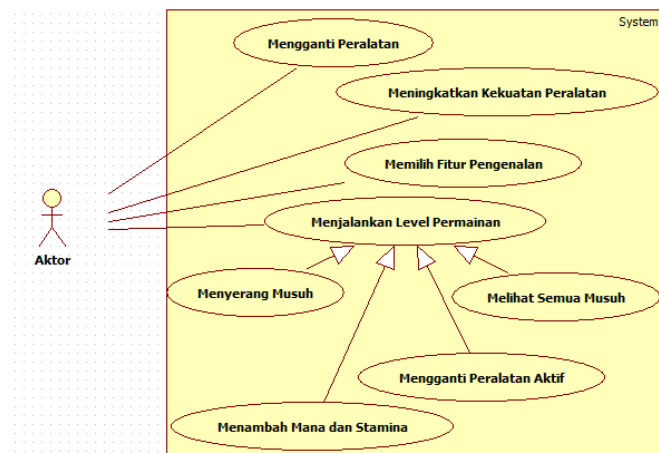
Peraturan permainan yang dibuat adalah:

1. Tujuan setiap *level* adalah mengalahkan semua musuh yang ada
2. Tidak ada batasan waktu dalam satu *level* permainan, namun saat musuh habis permainan akan berakhir
3. Pemain memiliki *Hit Point* (HP) yang akan berkurang bila terkena serangan, dan permainan akan berakhir bila *Hit Point* pemain mencapai 0.
4. Pemain memiliki *mana* dan *stamina* yang berkurang bila pemain menyerang musuh. *Mana* dan *stamina* yang habis bisa diisi lagi dengan menekan tombol *mana* dan *stamina*.
5. Setiap musuh akan bergerak mendekat lalu menyerang pemain.
6. Pemain akan menang apabila semua musuh telah terkalahkan, dan *level* berikutnya akan terbuka.

### C. Perancangan Sistem

Sistem yang dibangun pada Android adalah permainan yang berdasarkan *game mechanics* dan *dynamics* seperti yang telah disebutkan dan pengenalan gambar. Sistem pengenalan yang ada pada bagian Android adalah pengambilan gambar baru menggunakan kamera, penyesuaian *thresholding* gambar sesuai keinginan pemain dengan *OpenCv*, dan pengiriman gambar ke halaman *web* lokal dengan metode *POST*, dan menerima respon dari halaman *web* lokal.

Berdasarkan uraian permainan dan pengenalan gambar yang telah dijelaskan sebelumnya, terdapat beberapa spesifikasi kebutuhan fungsionalitas permainan yang harus dibuat, seperti pada Gambar 1.

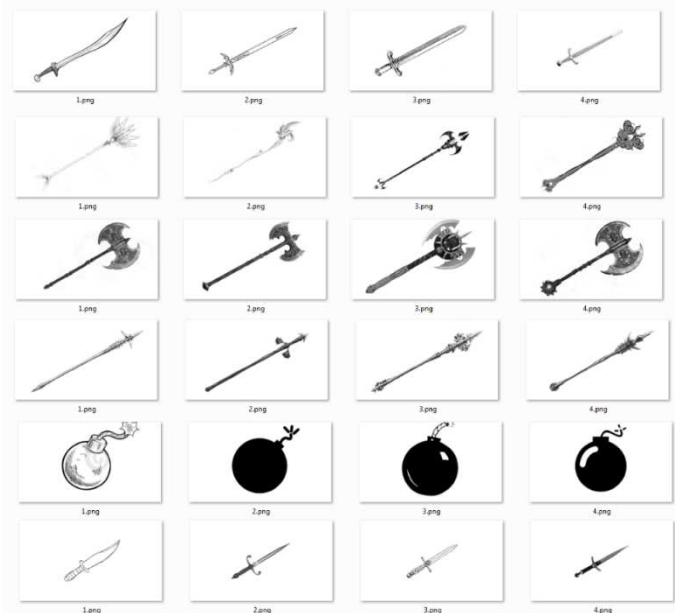


Gambar 1 Diagram Use Case

### D. Perancangan Dataset untuk Pengenalan

*Dataset* yang dibuat meliputi kategori peralatan yang akan dikenali, yaitu pedang, tongkat sihir, kapak, tombak, bom, dan pisau. Semua *dataset* gambar diambil dari *search engine* Google sebanyak 100 pada tiap kategori. Pada mulanya, ukuran tiap-tiap gambar diubah menjadi 480 x 270 *pixels*. Gambar kemudian dibuat hitam putih dan *background* dibersihkan sehingga menjadi putih. Contoh masing-masing *dataset* dapat

dilihat pada Gambar 2, masing-masing baris adalah satu kategori. Baris pertama adalah pedang, kemudian tongkat sihir, kapak, tombak, bom dan terakhir pisau.



Gambar 2. Contoh 6 kategori dataset

### E. Perancangan Pengenalan

Sistem pengenalan yang dibuat melalui 4 tahap yaitu pembuatan *dataset*, pembuatan model klasifikasi dengan *MATLAB*, pembuatan prediksi pengenalan data baru dengan *MATLAB*, dan mengubah kode prediksi pengenalan data baru menjadi *.Net assembly*.

Pembuatan model klasifikasi dilakukan dengan fungsi-fungsi *MATLAB*. Awalnya semua 6 kategori *dataset* diekstraksi fitur menggunakan *SURF*. Kemudian fitur tersebut diseleksi menggunakan *Bag of Words*. Fitur-fitur yang terpilih kemudian di klasifikasi menggunakan *SVM*. Hasil pada *workspace* *MATLAB* kemudian disimpan sebagai model klasifikasi.

Setelah model klasifikasi dibentuk, model dapat dimuat pada *workspace* baru. Kemudian data gambar baru dapat dimuat dan diprediksi dengan menggunakan fungsi dari *MATLAB*. Fungsi-fungsi ini dituliskan dalam bentuk kode.

Kode yang bertugas memprediksi data baru dapat diubah menjadi *.Net assembly* menggunakan fitur aplikasi *library compiler* pada *MATLAB*. Hasil dari aplikasi ini adalah sebuah *library* dengan ekstensi ".dll".

### F. Perancangan Halaman Web

Perancangan halaman *Web* menggunakan *ASP .Net* dengan bahasa *C#*. Halaman *web* dibangun dengan hanya dapat menerima metode *POST*. Halaman *web* menggunakan *library* tambahan yang berasal dari *MATLAB*. Halaman *web* menerima 3 jenis masukan yang kemudian di proses pada 3 bagian yang dijelaskan secara singkat sebagai berikut:

1. Pembuatan *user* baru ditujukan untuk membuat folder pada direktori lokal dengan nama *user* tersebut. Sistem menerima angka *random* sebagai *username*, lalu sistem mengecek apakah angka tersebut sudah ada dalam direktori lokal. Bila

belum ada maka sistem akan menyimpan angka sebagai *username* dengan membuat folder baru. Bila sudah ada sistem akan mengirim *response* untuk memilih ulang angka *random*.

2. Penyimpanan gambar ditujukan untuk menyimpan gambar pada direktori lokal. Sistem menerima *username* sebagai penentu letak folder yang akan diisi gambar.
3. Deteksi gambar ditujukan untuk mengenali gambar yang telah disimpan dalam direktori lokal. Sistem memproses gambar dengan memanggil fungsi dari *library* tambahan. Sistem kemudian mengirimkan *response* berisi hasil deteksi gambar.

#### IV. IMPLEMENTASI

Permainan dibuat menggunakan bahasa C# dengan aplikasi Unity 5.2.1fl. Fitur-fitur utama yang diimplementasikan dalam permainan adalah penyimpanan *data*, pergantian peralatan, pertandingan dalam sebuah *level*, pengambilan gambar dan pengiriman gambar.

Penyimpanan *data* adalah pembuatan *file* berisi *data progress* permainan pemain. Penyimpanan dan pemuatan *data* dilakukan secara otomatis.

Fitur pergantian peralatan berfungsi untuk mengganti peralatan yang akan dibawa pemain di dalam pertandingan. Peralatan yang dibawa dapat ditingkatkan kekuatannya dengan fitur tambahan yaitu meningkatkan kekuatan peralatan. Peralatan yang akan dibawa beserta tingkat kekuatannya disimpan dalam penyimpanan lokal.

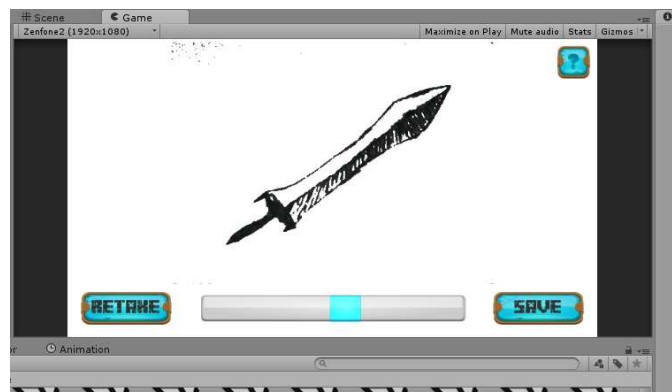
Fitur pertandingan dalam sebuah level membutuhkan fitur pemilihan level. Fitur pemilihan level bertugas untuk memberikan tingkat kesulitan bertahap pada pemain. Fitur pertandingan merupakan fitur paling kompleks dalam permainan ini. Di dalamnya dibutuhkan banyak fitur yang bekerja bersamaan. Fitur-fitur tersebut adalah: Fitur serangan jarak jauh dan serangan jarak dekat, fitur pergerakan musuh dan serangan musuh, fitur mana dan stamina, fitur mengganti peralatan aktif, dan fitur melihat semua musuh. Implementasi pertandingan dapat dilihat pada Gambar 3.

Fitur pengambilan gambar berfungsi untuk mengaktifkan kamera perangkat dan menunjukkannya pada layar. Dengan menekan sebuah tombol, maka gambar akan disimpan di penyimpanan lokal. Gambar yang telah disimpan kemudian diproses dengan fitur tambahan yaitu *Thresholding* sebelum gambar dikirim. Proses *thresholding* dapat dikontrol oleh pengguna untuk menghilangkan *background* sesuai kebutuhan. Implementasi *thresholding* dapat dilihat pada Gambar 4.

Fitur pengiriman gambar berfungsi untuk mengirimkan gambar ke halaman *web*. Sistem melakukan koneksi dengan halaman *web* sebanyak 3 kali, yang pertama untuk membuat user baru, yang kedua untuk mengirim gambar, dan yang terakhir untuk mengirim perintah deteksi gambar. Halaman *web* akan memberikan respon yang kemudian diolah oleh sistem.



Gambar 3. Pembuatan level permainan



Gambar 4. Tampilan proses *thresholding*

Pengenalan gambar dibuat dengan fungsi-fungsi dan aplikasi *library compiler* pada MATLAB. Fungsi pertama adalah fungsi penyimpanan *workspace* berisi model klasifikasi. Fungsi kedua adalah fungsi prediksi data baru dengan model klasifikasi sebelumnya. Fungsi kedua yang berupa kumpulan kode kemudian dibuat menjadi *.Net assembly* dengan aplikasi MATLAB.

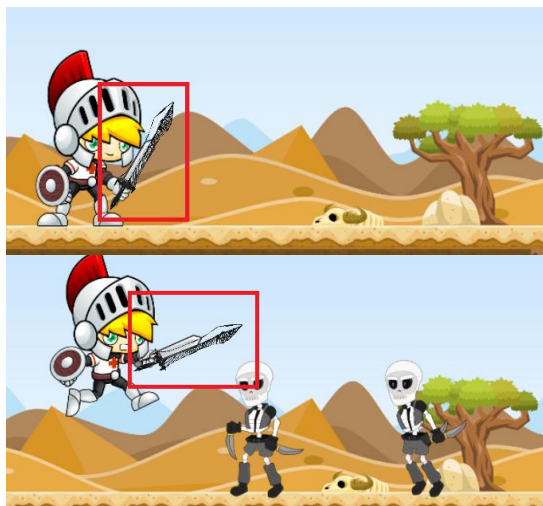
Pembuatan halaman *web* dilakukan dengan memproses 3 jenis input yaitu pembuatan *user*, penyimpanan gambar dan deteksi gambar. Halaman *web* menggunakan IIS x64 (*Internet Information Services*) dan “*Debug x64*” dan tambahan referensi dari library MATLAB.

#### V. PENGUJIAN DAN EVALUASI

Pengujian fungsionalitas dilakukan dengan menyiapkan sejumlah skenario sebagai tolok ukur keberhasilan pengujian. Pengujian fungsionalitas dengan mengacu pada kasus penggunaan dari aplikasi yang dikembangkan. Hasil pengujian fungsionalitas menunjukkan semua skenario pengujian berhasil dan program berjalan dengan baik. Sehingga bisa disimpulkan bahwa fungsionalitas dari program telah bisa bekerja sesuai dengan yang diharapkan. Hasil pengujian fungsionalitas dapat dilihat pada Tabel 1. Hasil uji coba integrasi gambar dapat dilihat pada Gambar 5.

Tabel 1.  
Pengujian Fungsionalitas

ID	Deskripsi	Hasil
UJ-UC-001	Uji Coba Pengenalan Gambar	Skenario 1 Berhasil
		Skenario 2 Berhasil
		Skenario 3 Berhasil
		Skenario 4 Berhasil
		Skenario 5 Berhasil
UJ-UC-002	Uji Coba Integrasi Gambar pada Objek Game	Skenario 1 Berhasil
		Skenario 2 Berhasil
UJ-UC-003	Uji Coba Aturan Main	Skenario 1 Berhasil
		Skenario 2 Berhasil
		Skenario 3 Berhasil
		Skenario 4 Berhasil
		Skenario 5 Berhasil
		Skenario 6 Berhasil
UJ-UC-004	Uji Coba Skenario Permainan	Skenario 1 Berhasil
		Skenario 2 Berhasil



Gambar 5. Uji coba integrasi gambar pada objek game

Pengujian non-fungsionalitas dilakukan dengan menyiapkan sejumlah skenario sebagai tolok ukur keberhasilan pengujian. Pengujian non-fungsionalitas dilakukan untuk mengetahui bagaimana hasil keluaran selain dari sisi fungsionalitas sistem terhadap skenario yang dipersiapkan. Hasil pengujian non-fungsionalitas menunjukkan bahwa permainan cukup menarik dan mudah dimengerti bagi pengguna baru. Hasil pengujian non-fungsionalitas dapat dilihat pada Tabel 2.

Tabel 2.  
Pengujian non-fungsionalitas

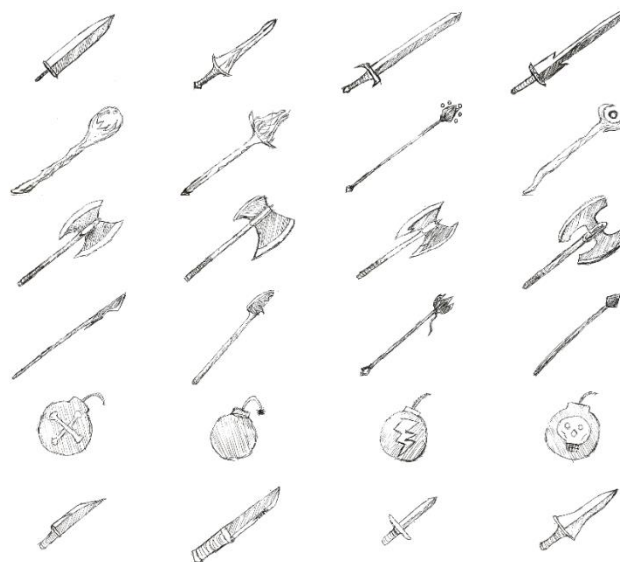
ID	Deskripsi	Rata-rata Nilai
UJ-UC-005	Uji Coba Daya Tarik Permainan	8.6
UJ-UC-006	Uji Coba Kejelasan Permainan	8.8

Pengujian akurasi pengenalan gambar dilakukan dengan menyiapkan 10 gambar pada masing- masing kategori peralatan sebagai tolok ukur keberhasilan pengujian. Pengujian akurasi pengenalan gambar dilakukan untuk mengetahui bagaimana tingkat keberhasilan sistem dalam mengenali gambar baru. Hasil pengujian akurasi pengenalan gambar menunjukkan tingkat pengenalan yang cukup rendah, hal ini mungkin disebabkan oleh jumlah dataset yang perlu diperbanyak.

Beberapa gambar yang di uji coba ditunjukkan pada Gambar 6. Hasil pengujian akurasi pengenalan gambar dapat dilihat pada Tabel 3.

Tabel 3.  
Pengujian akurasi pengenalan gambar

ID	Deskripsi	Rata-rata Nilai
UJ-UC-007	Uji Coba Akurasi Pengenalan Gambar	6.5



Gambar 6. Sebagian gambar yang dikenali

## VI. KESIMPULAN

Dari hasil uji coba yang telah dilakukan terhadap permainan dan sistem pengenalannya, dapat diambil kesimpulan sebagai berikut:

1. Menurut hasil uji coba kotak hitam terhadap uji coba fungsionalitas, aplikasi ini sudah memenuhi ekspektasi harapan penulis.
2. Menurut hasil uji coba daya tarik permainan, permainan telah memiliki daya tarik yang baik bagi pengguna.
3. Menurut uji coba kejelasan permainan, pengguna baru dapat dengan mudah mengerti alur permainan.
4. Menurut hasil uji coba pengenalan gambar, sistem belum dapat menentukan kategori suatu gambar dengan akurat.

## UCAPAN TERIMA KASIH

Penulis B.N. mengucapkan punji syukur terhadap Allah Subhanahu wa ta'ala yang telah melimpahkan kesehatan, semangat, dan hidayah-Nya sehingga penulis Alhamdulillah mampu menyelesaikan penelitian dengan baik. Penulis juga mengucapkan terima kasih kepada Bapak Imam Kuswardayan dan Ibu Nanik Suciati yang telah banyak membantu penulis dalam menyelesaikan penelitian ini.

#### DAFTAR PUSTAKA

- [1] M. Simkova, "Using of Computer Games in Supporting Education," *Procedia - Social and Behavioral Sciences*, vol. 141, pp. 1224-1227, 2014.
- [2] P. Sykora, P. Kamencay and R. Hudec, "Comparison of SIFT and SURF Methods for Use on Hand Gesture Recognition based on Depth Map," *AASRI Procedia*, vol. 9, pp. 19-24, 2014.
- [3] W. Chong and H. Kaiqi, "How to use Bag-of-Words model better for image classification," *Image and Vision Computing*, vol. 38, pp. 65-74, 2015.
- [4] M. Slavomir, H. Robert, K. Patrik, B. Miroslav and Z. Martina, "Classification of Wild Animals based on SVM and Local Descriptors," *AASRI Procedia*, vol. 9, pp. 25-30, 2014.
- [5] T. T. Cota, L. Ishitania and N. Vieira Jr., "Mobile game design for the elderly: A study with focus on the motivation to play," *Computers in Human Behavior*, vol. 51, pp. 96-105, 2015.