

RANCANG BANGUN APLIKASI *AUGMENTED REALITY* BERBASIS ANDROID MENGGUNAKAN ALGORITMA *ORIENTED FAST AND ROTATED BRIEF* (ORB) SEBAGAI MEDIA EDUKASI PENGENALAN KOMPONEN *HARDWARE* KOMPUTER

Syaikhah Rabbani Rofid¹, Al-Khowarizmi²

^{1,2}Program Studi Teknologi Informasi,
Fakultas Ilmu Komputer dan Teknologi Informasi,
Universitas Muhammadiyah Sumatera Utara, Indonesia

Email: rabbanirofid1406@gmail.com¹, alkhowarizmi@umsu.ac.id²

ABSTRAK

Penelitian ini bertujuan untuk merancang dan membangun aplikasi *Augmented Reality* (AR) berbasis Android sebagai media edukasi pengenalan komponen *hardware* komputer dengan mengimplementasikan algoritma *Oriented FAST and Rotated BRIEF* (ORB). Penelitian ini dilatarbelakangi oleh masih terbatasnya media pembelajaran interaktif yang mampu membantu pengguna dalam mengenal bentuk dan fungsi komponen *hardware* komputer secara lebih menarik dan mudah dipahami.

Metode penelitian yang digunakan adalah *Research and Development* (R&D) dengan model pengembangan *Prototype*, yang meliputi tahapan komunikasi kebutuhan, perancangan cepat (*quick design*), pembangunan *prototype*, evaluasi pengguna, serta penyempurnaan sistem. Aplikasi dikembangkan menggunakan Android Studio sebagai lingkungan pengembangan, OpenCV SDK sebagai pustaka *computer vision*, *Android Native Development Kit* (NDK) untuk implementasi algoritma ORB menggunakan bahasa C++, serta objek tiga dimensi berformat .glb yang diintegrasikan ke dalam aplikasi.

Hasil penelitian menunjukkan bahwa aplikasi mampu mendeteksi *marker* menggunakan algoritma ORB dan menampilkan objek tiga dimensi beserta informasi komponen *hardware* komputer secara *real-time* pada perangkat Android. Pengujian aplikasi dilakukan menggunakan metode *Black Box Testing* dan menunjukkan bahwa seluruh fungsi utama aplikasi berjalan sesuai dengan kebutuhan fungsional yang telah dirancang. Dengan demikian, aplikasi yang dikembangkan dapat digunakan sebagai media edukasi yang interaktif dalam pengenalan komponen *hardware* komputer.

Kata kunci: *Augmented Reality*; Android; OpenCV; *Oriented FAST and Rotated BRIEF* (ORB); *Hardware Komputer*; Media Edukasi

ABSTRACT

This research aims to design and develop an Android-based Augmented Reality (AR) application as an educational medium for introducing computer hardware components by implementing the Oriented FAST and Rotated BRIEF (ORB) algorithm. The study is motivated by the limited availability of interactive learning media that can effectively help users understand the shapes and functions of computer hardware components in a more engaging and interactive manner.

This research employed the Research and Development (R&D) method using the Prototype development model, which consists of communication, quick design, prototype construction, user evaluation, and system refinement. The application was developed using Android Studio as the development environment, OpenCV SDK as the

computer vision library, and the Android Native Development Kit (NDK) to implement the ORB algorithm in C++. Three-dimensional (3D) objects in .glb format were integrated into the application to visualize computer hardware components.

The results show that the application is capable of detecting image markers using the ORB algorithm and displaying three-dimensional objects along with information about computer hardware components in real time on Android devices. System testing was conducted using the Black Box Testing method, and the results indicate that all main application functions operated as expected without significant functional errors. Therefore, the developed application can serve as an interactive educational medium for introducing computer hardware components.

Keywords: *Augmented Reality; Android; OpenCV; Oriented FAST and Rotated BRIEF (ORB); Computer Hardware; Educational Media*

PENDAHULUAN

Perkembangan teknologi digital telah membawa perubahan yang signifikan dalam dunia pendidikan, khususnya dalam penggunaan media pembelajaran yang memanfaatkan teknologi informasi. Kondisi tersebut mendorong pendidik untuk beralih dari pola penyampaian materi yang bersifat konvensional menuju pemanfaatan teknologi interaktif yang dapat meningkatkan keterlibatan dan pemahaman peserta didik. Di Indonesia, penggunaan perangkat mobile berbasis Android terus mengalami peningkatan, terutama di kalangan pelajar. Perkembangan tersebut membuka peluang yang semakin besar untuk mengembangkan media pembelajaran digital yang inovatif, fleksibel, dan dapat diakses secara mudah. (Statistik, 2023).

Perkembangan teknologi digital telah mendorong munculnya berbagai inovasi dalam penyampaian informasi dan proses pembelajaran melalui pemanfaatan media edukasi berbasis teknologi. Penggunaan media pembelajaran interaktif dapat memudahkan pengguna dalam memahami materi karena informasi disajikan secara visual, menarik, dan mudah dipahami, serta memungkinkan adanya interaksi langsung dengan materi yang dipelajari. Selain itu, integrasi teknologi digital dalam media edukasi dapat meningkatkan keterlibatan pengguna dan menciptakan pengalaman belajar yang lebih menarik dibandingkan dengan pendekatan pembelajaran yang masih bersifat konvensional. (Krishnamurthy & Kabilamani, 2024; Santi et al., 2021).

Pada jenjang Sekolah Menengah Pertama (SMP), mata pelajaran Teknologi Informasi dan Komunikasi (TIK) atau Informatika berperan dalam memberikan pemahaman dasar mengenai teknologi, termasuk pengenalan komponen *hardware* komputer, bentuk, dan fungsinya. Meskipun demikian, pembelajaran pada materi tersebut masih menghadapi sejumlah kendala, antara lain keterbatasan fasilitas pembelajaran, minimnya visualisasi objek secara langsung, serta penggunaan buku dan gambar dua dimensi yang masih dominan. Kondisi tersebut dapat memengaruhi tingkat pemahaman siswa dalam mengenali perangkat keras komputer secara lebih konkret (Djafar & Novian, 2021).

Salah satu teknologi yang berpotensi mendukung terciptanya pembelajaran visual yang interaktif adalah *Augmented Reality* (AR). Teknologi ini memungkinkan

objek virtual tiga dimensi diintegrasikan dengan lingkungan nyata secara langsung melalui kamera pada perangkat mobile. Dalam konteks pendidikan, pemanfaatan AR dapat menciptakan pengalaman belajar yang lebih imersif karena siswa dapat mengamati sekaligus berinteraksi dengan objek pembelajaran yang ditampilkan seolah-olah berada di lingkungan nyata. Hasil penelitian (Imam et al., 2021) menunjukkan bahwa penggunaan AR sebagai media pembelajaran berkontribusi terhadap peningkatan pemahaman konsep dan minat siswa terhadap materi yang dipelajari.

Pengembangan aplikasi AR pada perangkat Android pada umumnya membutuhkan penerapan teknik *computer vision* untuk mengenali *marker* atau citra tertentu yang digunakan sebagai pemicu dalam menampilkan objek virtual. Salah satu algoritma yang dapat digunakan untuk mendeteksi dan mencocokkan fitur pada citra adalah *Oriented FAST and Rotated BRIEF* (ORB). Algoritma ORB memiliki keunggulan dalam aspek kecepatan pemrosesan dan efisiensi komputasi, sehingga relevan untuk diterapkan pada perangkat mobile dengan sumber daya yang relatif terbatas. Menurut (Sumardi, 2021), penggunaan algoritma ORB dapat mendukung proses pendeteksian objek yang stabil dan responsif pada aplikasi AR berbasis Android.

Sejumlah penelitian sebelumnya telah mengeksplorasi pengembangan aplikasi *Augmented Reality* berbasis Android sebagai media edukasi dengan memanfaatkan teknologi *computer vision*. Penelitian (Dinara, Zykov, et al., 2023) menunjukkan bahwa OpenCV dapat digunakan dalam pengembangan aplikasi AR yang mendukung proses pengenalan objek serta pengolahan citra secara *real-time*. Penelitian lainnya oleh (Li et al., 2024) mengembangkan platform AR berbasis Android dengan menerapkan algoritma *Oriented FAST and Rotated BRIEF* (ORB) untuk mendukung proses pendeteksian fitur pada objek virtual. Namun, sejumlah penelitian tersebut masih banyak memanfaatkan *framework* atau *library* AR tertentu. Sementara itu, implementasi algoritma ORB secara langsung melalui OpenCV SDK pada Android Studio, khususnya untuk proses deteksi *marker* dan *feature matching*, masih relatif terbatas. Berdasarkan kondisi tersebut, penelitian ini berfokus pada perancangan dan pengembangan aplikasi *Augmented Reality* berbasis Android dengan mengimplementasikan algoritma ORB melalui OpenCV SDK pada Android Studio sebagai metode utama untuk mendeteksi *marker* serta menampilkan objek tiga dimensi secara *real-time*.

Sejumlah penelitian terdahulu menunjukkan bahwa pengembangan aplikasi *Augmented Reality* (AR) dalam bidang pendidikan masih menghadapi sejumlah keterbatasan. Penelitian yang telah dilakukan umumnya menitikberatkan pada pemanfaatan teknologi AR secara luas dan belum secara spesifik diarahkan untuk menunjang pembelajaran mengenai pengenalan komponen *hardware* komputer pada tingkat Sekolah Menengah Pertama (SMP). Di sisi lain, penerapan algoritma *Oriented FAST and Rotated BRIEF* (ORB) pada aplikasi AR berbasis Android yang dikembangkan melalui Android Studio dengan dukungan OpenCV juga masih belum banyak diterapkan. OpenCV memiliki beragam fungsi *computer vision* yang dapat

digunakan untuk proses deteksi serta pencocokan fitur secara *real-time*, sehingga berpotensi mendukung peningkatan performa aplikasi AR pada perangkat Android (Yuhano & Akbar, 2024).

Berdasarkan kondisi tersebut, penelitian ini diarahkan pada perancangan dan pembangunan aplikasi *Augmented Reality* berbasis Android dengan menerapkan algoritma *Oriented FAST and Rotated BRIEF* (ORB) melalui pustaka OpenCV pada lingkungan Android Studio. Aplikasi yang dikembangkan ditujukan sebagai media pembelajaran interaktif untuk membantu siswa Sekolah Menengah Pertama (SMP) mengenali berbagai komponen *hardware* komputer. Sistem memanfaatkan *marker* berbasis citra sebagai pemicu, sehingga model objek tiga dimensi dapat ditampilkan secara *real-time* setelah *marker* berhasil teridentifikasi oleh kamera perangkat. Melalui pengembangan tersebut, penelitian ini diharapkan dapat menghasilkan alternatif media pembelajaran berbasis *Augmented Reality* yang interaktif dan dapat menunjang proses pembelajaran dalam mengenali komponen *hardware* komputer.

METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Research and Development* (R&D) dengan model pengembangan *Prototype* untuk menghasilkan aplikasi *Augmented Reality* (AR) berbasis Android sebagai media edukasi pengenalan komponen *hardware* komputer bagi siswa Sekolah Menengah Pertama (SMP). Model *Prototype* dipilih karena memungkinkan proses pengembangan dilakukan secara bertahap melalui identifikasi kebutuhan, perencanaan, perancangan, implementasi, pengujian, dan penyempurnaan produk. Tahapan pengembangan meliputi *communication*, *quick plan*, *modeling quick design*, *construction of prototype*, dan *deployment*. Pada tahap *communication*, dilakukan identifikasi kebutuhan pengguna, kebutuhan perangkat keras dan perangkat lunak, serta materi komponen *hardware* komputer. Tahap *quick plan* digunakan untuk menetapkan kebutuhan fungsional dan nonfungsional serta merancang alur kerja aplikasi. Selanjutnya, *modeling quick design* mencakup perancangan antarmuka, navigasi, *marker*, dan objek tiga dimensi (3D). Tahap *construction of prototype* dilakukan melalui pembangunan aplikasi menggunakan Android Studio, implementasi algoritma *Oriented FAST and Rotated BRIEF* (ORB) dengan OpenCV, serta integrasi *marker*, objek 3D, dan fitur aplikasi. Tahap *deployment* dilakukan melalui pengujian fungsional dan penyempurnaan aplikasi berdasarkan hasil pengujian.

Perancangan sistem dilakukan sebagai dasar implementasi aplikasi dengan menggunakan *Use Case Diagram*, *flowchart*, perancangan antarmuka, *marker* berbasis citra, dan objek 3D. Aplikasi dikembangkan menggunakan Android Studio dengan bahasa Java untuk membangun antarmuka dan mengelola logika aplikasi, sedangkan proses pengolahan citra dan pendeteksian *marker* menggunakan OpenCV dan algoritma ORB yang diimplementasikan melalui C++ dengan Android Native Development Kit (NDK). Integrasi antara Java dan C++ dilakukan menggunakan *Java*

Native Interface (JNI). Kamera perangkat Android digunakan untuk menangkap citra secara *real-time*, kemudian citra diproses melalui OpenCV untuk mendeteksi *keypoint*, mengekstraksi deskriptor, dan melakukan *feature matching* terhadap *marker* yang telah disimpan sebagai data referensi. Ketika *marker* berhasil dikenali, sistem menentukan posisi dan orientasinya terhadap kamera untuk selanjutnya menampilkan objek 3D beserta informasi komponen *hardware* komputer yang sesuai. Objek 3D diperoleh dari platform Sketchfab dalam format GLB dan disesuaikan dengan kebutuhan aplikasi sebelum diintegrasikan dengan *marker*. Perangkat pengembangan menggunakan laptop Acer Nitro AN515-512 dengan prosesor Intel Core i5-8300H, RAM 8 GB, penyimpanan 1,14 TB, dan sistem operasi Windows 11, sedangkan aplikasi ditujukan untuk perangkat Android dengan RAM minimal 2 GB, Android versi 8.0 atau lebih baru, dan kamera belakang.

Pengujian produk dilakukan menggunakan metode *Black Box Testing* untuk mengevaluasi kesesuaian fungsi aplikasi berdasarkan hubungan antara masukan (*input*) dan keluaran (*output*) tanpa memperhatikan struktur kode program. Subjek pengujian terdiri atas satu orang guru dan 30 siswa SMP yang menggunakan aplikasi secara langsung, sedangkan objek pengujian adalah aplikasi AR berbasis Android untuk pengenalan komponen *hardware* komputer. Instrumen pengujian berupa lembar skenario *Black Box Testing* yang memuat fitur yang diuji, skenario pengujian, hasil yang diharapkan, hasil aktual, dan status *Pass* atau *Fail*. Data dikumpulkan melalui observasi langsung selama pelaksanaan pengujian dengan mencatat keberhasilan setiap fungsi, kesesuaian hasil aktual dengan hasil yang diharapkan, serta temuan berupa *bug*, kegagalan sistem, atau ketidaksesuaian keluaran. Analisis data dilakukan secara deskriptif dengan membandingkan hasil aktual dan hasil yang diharapkan pada setiap skenario. Persentase keberhasilan dihitung berdasarkan perbandingan jumlah skenario berstatus *Pass* dengan seluruh skenario pengujian. Aplikasi dinyatakan memenuhi kebutuhan fungsional apabila seluruh skenario memperoleh status *Pass*, yang ditunjukkan melalui persentase keberhasilan 100%. Pengujian mencakup pembukaan dan penutupan aplikasi, akses menu, aktivasi kamera, deteksi *marker* menggunakan ORB melalui OpenCV, penampilan objek 3D secara *real-time*, serta kesesuaian informasi komponen *hardware* yang ditampilkan.

HASIL DAN PEMBAHASAN

Hasil Pengembangan Aplikasi

Penelitian ini menghasilkan aplikasi *Augmented Reality* (AR) berbasis Android yang dikembangkan sebagai media edukasi untuk membantu siswa Sekolah Menengah Pertama (SMP) mengenali komponen *hardware* komputer. Aplikasi dikembangkan menggunakan metode *Research and Development* (R&D) dengan model *Prototype*. Tahapan pengembangan meliputi identifikasi kebutuhan, perancangan sistem dan antarmuka, pembuatan prototipe, implementasi algoritma *Oriented FAST and Rotated BRIEF* (ORB), serta pengujian aplikasi. Implementasi dilakukan menggunakan Android Studio dengan bahasa pemrograman Java untuk membangun

antarmuka dan mengatur alur aplikasi, sedangkan proses pengolahan citra dan deteksi marker menggunakan OpenCV melalui bahasa C++ yang dikompilasi dengan Android Native Development Kit (NDK). Komunikasi antara modul Java dan C++ dilakukan melalui *Java Native Interface* (JNI), sehingga hasil deteksi dapat diteruskan ke aplikasi Android untuk mendukung visualisasi objek secara *real-time*.

Aplikasi menyediakan sepuluh komponen *hardware* komputer yang dapat dikenali melalui marker, yaitu Monitor, Mouse, Keyboard, Casing CPU, Processor, Motherboard, Hard Disk Drive (HDD), Power Supply, Heatsink, dan RAM. Setiap marker dihubungkan dengan objek tiga dimensi (3D) serta informasi mengenai komponen yang bersesuaian. Ketika marker dikenali oleh kamera, aplikasi menampilkan objek 3D dan informasi komponen pada halaman *Scan Marker*. Selain visualisasi, objek 3D dapat diputar dan diperbesar atau diperkecil menggunakan interaksi sentuhan. Aplikasi juga menyediakan fungsi *Reset* untuk menghapus objek yang sedang ditampilkan dan mengembalikan sistem ke kondisi pencarian marker. Antarmuka aplikasi terdiri atas halaman utama, *Scan Marker*, Info, Cara Penggunaan, dan Informasi Pengembang. Halaman *Scan Marker* menjadi bagian utama karena digunakan untuk mengaktifkan kamera dan melakukan pendeteksian marker secara *real-time*. Sementara itu, halaman Info menyediakan akses terhadap marker, petunjuk penggunaan, dan informasi pengembang. Struktur antarmuka tersebut dirancang sederhana dengan navigasi yang jelas agar fungsi aplikasi dapat diakses secara sistematis oleh pengguna.

Penggunaan aplikasi oleh siswa menunjukkan bahwa proses pembelajaran dapat dilakukan melalui interaksi langsung dengan marker. Pengguna mengarahkan kamera perangkat Android ke marker yang telah disediakan, kemudian sistem mengenali marker dan menampilkan objek 3D sesuai dengan komponen yang terdaftar. Objek tersebut dapat diamati dari berbagai sudut melalui fitur rotasi dan *zoom*. Hasil ini menunjukkan bahwa aplikasi tidak hanya menyediakan informasi tekstual, tetapi juga menghadirkan representasi visual komponen *hardware* dalam bentuk tiga dimensi sehingga memberikan alternatif penyajian materi yang lebih interaktif.



Gambar 1 Hasil Rancang Bangun Halaman Utama Aplikasi

Hasil Rancang Bangun Aplikasi

Hasil dari penelitian ini berupa sebuah aplikasi *Augmented Reality* (AR) berbasis Android yang dikembangkan sebagai media edukasi pengenalan komponen *hardware* komputer untuk siswa Sekolah Menengah Pertama (SMP). Aplikasi dirancang untuk membantu proses pembelajaran dengan memanfaatkan teknologi *Augmented Reality* sehingga pengguna dapat melihat visualisasi objek tiga dimensi (3D) komponen *hardware* komputer secara interaktif melalui kamera pada perangkat Android.

Proses rancang bangun aplikasi dilakukan berdasarkan metode *Research and Development* (R&D) dengan model *Prototype* sebagaimana telah dijelaskan pada Bab III. Pengembangan aplikasi dilaksanakan secara bertahap mulai dari identifikasi kebutuhan pengguna, perancangan sistem, perancangan antarmuka, pembuatan prototipe, penerapan algoritma *Oriented FAST and Rotated BRIEF* (ORB), hingga pengujian aplikasi untuk memastikan seluruh fungsi berjalan sesuai dengan kebutuhan yang telah ditetapkan.

Aplikasi dikembangkan menggunakan Android Studio sebagai lingkungan pengembangan utama dengan bahasa pemrograman Java untuk membangun antarmuka dan mengelola alur aplikasi. Sementara itu, proses pengolahan citra dan pendeteksian *marker* diimplementasikan menggunakan pustaka OpenCV melalui bahasa pemrograman C++ yang dikompilasi menggunakan *Android Native Development Kit* (NDK). Komunikasi antara modul Java dan C++ dilakukan melalui *Java Native Interface* (JNI), sehingga proses pendeteksian *marker* dapat dilakukan secara efisien dan hasilnya dapat digunakan untuk menampilkan objek tiga dimensi secara *real-time*.

Pada aplikasi yang dikembangkan, setiap *marker* telah dihubungkan dengan objek tiga dimensi komponen *hardware* komputer yang sesuai. Ketika kamera perangkat Android diarahkan ke *marker*, sistem akan melakukan proses pengambilan

citra, mendeteksi titik-titik fitur menggunakan algoritma ORB, melakukan pencocokan dengan data *marker* yang tersimpan pada aplikasi, kemudian menampilkan objek tiga dimensi beserta informasi mengenai komponen *hardware* komputer yang berhasil dikenali. Dengan mekanisme tersebut, pengguna dapat mempelajari bentuk dan mengenali komponen *hardware* komputer secara lebih interaktif dibandingkan media pembelajaran konvensional.

Secara umum, aplikasi yang dihasilkan memiliki beberapa fitur utama, yaitu menu *Scan Marker* untuk memulai proses pendeteksian *marker*, menu Info yang berisi petunjuk penggunaan aplikasi, biodata pengembang, serta fasilitas untuk mengunduh *marker*, dan menu Keluar untuk menutup aplikasi. Seluruh fitur tersebut dirancang agar mudah digunakan oleh siswa SMP sehingga dapat mendukung proses pembelajaran pengenalan komponen *hardware* komputer secara efektif.

Hasil rancang bangun aplikasi ini menjadi dasar untuk pembahasan pada subbab berikutnya, yaitu hasil rancang bangun antarmuka aplikasi, penerapan algoritma ORB dalam proses pendeteksian *marker*, serta hasil pengujian aplikasi yang dilakukan untuk mengetahui kesesuaian fungsi aplikasi dengan kebutuhan yang telah dirancang.

Implementasi ORB untuk Deteksi Marker

ORB diterapkan sebagai mekanisme utama dalam proses pengenalan marker. Proses deteksi dimulai dari pengambilan *frame* kamera secara *real-time*, kemudian citra dipersiapkan untuk proses ekstraksi fitur. Pada citra marker yang telah didaftarkan, format RGBA dikonversi menjadi *grayscale* menggunakan fungsi `cv::cvtColor()`, sedangkan *frame* kamera yang diperoleh melalui *Y-buffer* CameraX telah direpresentasikan sebagai citra satu kanal. Selanjutnya, sistem melakukan ekstraksi *keypoints* dan *descriptor* menggunakan ORB. Pada implementasinya, detektor ORB dikonfigurasi dengan maksimum 1.500 fitur melalui `cv::ORB::create(1500)` dan proses ekstraksi dilakukan menggunakan `detectAndCompute()`.

Descriptor dari marker referensi kemudian dibandingkan dengan descriptor yang diperoleh dari *frame* kamera menggunakan *Brute-Force Hamming*. Proses pencocokan dilakukan dengan `knnMatch()` untuk memperoleh dua kandidat kecocokan terdekat. Hasil tersebut selanjutnya disaring menggunakan *ratio test* dengan nilai 0,75 untuk memperoleh *good matches*. Sistem menetapkan jumlah minimum 15 *good matches* sebagai salah satu syarat sebelum proses validasi dilanjutkan. Mekanisme tersebut digunakan untuk mengurangi kecocokan fitur yang kurang sesuai sebelum marker dinyatakan terdeteksi.

Validasi hasil pencocokan dilakukan secara bertahap menggunakan beberapa kriteria. Setelah jumlah *good matches* memenuhi batas minimum, sistem menghitung *homography* menggunakan metode RANSAC dengan ambang kesalahan 5,0. Hasil tersebut kemudian diperiksa berdasarkan jumlah *inlier*, dengan minimum 10 *inlier*. Sistem juga melakukan pemeriksaan bentuk kontur menggunakan `cv::isContourConvex()` serta luas area menggunakan `cv::contourArea()`. Area

dinyatakan memenuhi kriteria apabila berbentuk cembung dan memiliki luas minimum 2.000 piksel serta tidak melampaui batas maksimum berdasarkan ukuran *frame*.

Apabila seluruh kriteria terpenuhi, sistem menentukan posisi dan orientasi marker terhadap kamera menggunakan `solvePnP()` dengan metode `SOLVEPNP_ITERATIVE`. Proses ini menggunakan *object points* dan *image points* yang berasal dari titik fitur yang telah divalidasi. Hasil `solvePnP()` berupa vektor translasi (*tvec*) dan rotasi (*rvec*), yang selanjutnya digunakan untuk memperoleh posisi dan orientasi marker sebelum diteruskan ke aplikasi Android.

Implementasi tersebut menunjukkan bahwa ORB dalam penelitian ini tidak digunakan secara tunggal, tetapi dipadukan dengan proses *feature matching*, *ratio test*, RANSAC, pemeriksaan *inlier*, validasi kontur, dan `solvePnP()`. Rangkaian proses tersebut memberikan mekanisme validasi berlapis sebelum sistem menampilkan objek 3D. Dengan demikian, hasil deteksi tidak hanya didasarkan pada kemiripan fitur, tetapi juga mempertimbangkan konsistensi geometris dan posisi marker pada citra kamera.

Hasil Deteksi Marker dan Visualisasi Objek 3D

Pengujian terhadap marker dilakukan pada sepuluh marker yang masing-masing merepresentasikan komponen *hardware* komputer. Hasil pengujian menunjukkan bahwa seluruh marker dapat dikenali sesuai dengan data yang telah didaftarkan pada sistem.

No.	Marker	Komponen yang Diharapkan	Hasil Pendeteksian	Status
1	Marker Monitor	Monitor	Monitor terdeteksi	Berhasil
2	Marker Mouse	Mouse	Mouse terdeteksi	Berhasil
3	Marker Keyboard	Keyboard	Keyboard terdeteksi	Berhasil
4	Marker Casing CPU	Casing CPU	Casing CPU terdeteksi	Berhasil
5	Marker Processor	Processor	Processor terdeteksi	Berhasil
6	Marker Motherboard	Motherboard	Motherboard terdeteksi	Berhasil
7	Marker Hard Disk Drive	Hard Disk Drive	Hard Disk Drive terdeteksi	Berhasil
8	Marker Power Supply	Power Supply	Power Supply terdeteksi	Berhasil
9	Marker Heatsink	Heatsink	Heatsink terdeteksi	Berhasil
10	Marker RAM	RAM	RAM terdeteksi	Berhasil

Hasil tersebut menunjukkan bahwa kesepuluh marker berhasil diidentifikasi sesuai dengan komponen yang diharapkan. Marker Monitor dikenali sebagai Monitor, Mouse sebagai Mouse, Keyboard sebagai Keyboard, dan hasil yang sama diperoleh pada Casing CPU, Processor, Motherboard, Hard Disk Drive, Power Supply, Heatsink,

serta RAM. Keberhasilan identifikasi tersebut menunjukkan bahwa kombinasi ORB dan mekanisme validasi yang diterapkan mampu digunakan untuk mengenali marker yang telah didaftarkan dalam aplikasi.

Keberhasilan deteksi marker selanjutnya digunakan untuk menentukan objek 3D yang ditampilkan. Setiap marker memiliki pasangan objek 3D tertentu sehingga hasil identifikasi marker secara langsung menentukan model yang ditampilkan pada halaman *Scan Marker*. Selain objek 3D, aplikasi juga menampilkan nama dan informasi mengenai komponen yang dikenali. Dengan demikian, proses pembelajaran tidak berhenti pada identifikasi marker, tetapi dilanjutkan dengan penyajian representasi visual komponen yang dapat diamati melalui kamera perangkat Android.

Fitur interaksi memungkinkan pengguna melakukan rotasi objek dengan gerakan geser serta melakukan *zoom in* dan *zoom out*. Fitur ini memberikan kesempatan kepada pengguna untuk mengamati objek dari berbagai sudut dan memperbesar bagian tertentu dari komponen. Berdasarkan penggunaan aplikasi oleh siswa, objek 3D dapat diamati secara langsung melalui layar perangkat sehingga informasi mengenai bentuk komponen dapat disajikan bersamaan dengan informasi tekstual.

Namun, hasil implementasi juga menunjukkan adanya keterbatasan pada mekanisme pelacakan objek. Setelah marker berhasil dikenali dan objek 3D ditampilkan, objek belum mengikuti perubahan posisi marker secara kontinu ketika marker digeser dari area *frame* kamera. Objek tetap berada pada posisi terakhir sampai pengguna melakukan tindakan tertentu, seperti menekan tombol *Reset*. Kondisi ini menunjukkan bahwa implementasi yang digunakan telah mendukung pengenalan marker dan penampilan objek 3D, tetapi belum menerapkan *continuous tracking* secara penuh.



Gambar 2 Penampilan Objek 3D Setelah *Marker* Berhasil Dideteksi

Berdasarkan Gambar diatas objek 3D ditampilkan setelah sistem berhasil mengenali *marker* yang sesuai. Sebagai contoh, ketika *marker Keyboard* berhasil dikenali, aplikasi menampilkan model 3D *Keyboard* beserta informasi mengenai komponen tersebut. Proses yang sama diterapkan pada komponen *hardware* lainnya,

yaitu *Monitor, Mouse, Casing CPU, Processor, Motherboard, Hard Disk Drive (HDD), Power Supply, Heatsink, dan RAM.*

Hasil Pengujian Fungsional

Pengujian fungsional dilakukan menggunakan metode *Black Box Testing* untuk mengetahui kesesuaian fungsi aplikasi dengan kebutuhan yang telah ditentukan. Pengujian dilakukan dengan memberikan masukan atau tindakan pada fitur tertentu dan mengamati keluaran yang dihasilkan tanpa melihat proses internal program. Sebanyak sembilan skenario diuji, meliputi pembukaan aplikasi, menu Info, Petunjuk, Biodata, aktivasi kamera, deteksi marker ORB, penampilan objek 3D, informasi *hardware*, dan fungsi keluar aplikasi.

No.	Fungsi yang Diuji	Hasil
1	Membuka aplikasi	Pass
2	Menu Info	Pass
3	Menu Petunjuk	Pass
4	Menu Biodata	Pass
5	Mengaktifkan kamera	Pass
6	Deteksi marker ORB	Pass
7	Menampilkan objek 3D	Pass
8	Menampilkan informasi hardware	Pass
9	Keluar aplikasi	Pass

Seluruh sembilan skenario memperoleh status *Pass*. Persentase keberhasilan pengujian dihitung berdasarkan perbandingan jumlah skenario yang berhasil dengan jumlah seluruh skenario, sehingga diperoleh tingkat keberhasilan sebesar 100%. Hasil tersebut menunjukkan bahwa seluruh fungsi utama yang diuji telah berjalan sesuai dengan keluaran yang diharapkan dan kebutuhan fungsional yang ditetapkan.

Hasil pengujian fungsional tersebut memperlihatkan bahwa integrasi antara antarmuka aplikasi, kamera, proses deteksi marker, visualisasi objek 3D, penyajian informasi, dan navigasi aplikasi dapat berjalan dalam satu alur penggunaan. Temuan ini memperkuat hasil pengujian marker yang menunjukkan keberhasilan identifikasi sepuluh komponen. Meskipun demikian, keberhasilan *Black Box Testing* perlu dipahami dalam konteks skenario yang diuji, karena pengujian tersebut terutama mengukur kesesuaian fungsi berdasarkan masukan dan keluaran, bukan pengukuran kuantitatif terhadap akurasi deteksi atau performa *tracking* secara kontinu.

Pengujian pada Perangkat Android

Pengujian pada perangkat Android dilakukan untuk mengetahui kemampuan aplikasi ketika dijalankan secara langsung pada perangkat yang digunakan dalam penelitian. Pengujian ini berfokus pada kondisi aplikasi selama digunakan, meliputi proses menjalankan aplikasi, penggunaan kamera untuk pendeteksian *marker*,

penampilan objek 3D, serta penggunaan fitur interaksi terhadap objek. Pengujian dilakukan pada perangkat Android yang digunakan sebagai perangkat uji untuk memastikan aplikasi dapat berjalan sesuai dengan kebutuhan yang telah ditentukan.

Tabel 3 Spesifikasi Perangkat Android yang Digunakan

No	Spesifikasi	Keterangan
1	Merek/Model	OPPO Reno11 5G
2	Sistem Operasi	ColorOS 16
3	RAM	8,00 GB
4	Kamera	Depan 32MP. Belakang 50MP+32MP+8MP
5	Penyimpanan Internal	256 GB

Berdasarkan pengujian yang dilakukan pada perangkat Android tersebut, aplikasi dapat dijalankan dengan baik. Kamera dapat digunakan untuk mengambil citra secara *real-time* sebagai masukan proses pendeteksian *marker*. Ketika *marker* yang sesuai diarahkan ke kamera, sistem dapat mengenalinya dan menampilkan objek 3D beserta informasi komponen *hardware* yang sesuai. Objek 3D juga dapat diberikan interaksi berupa rotasi dan pembesaran atau pengecilan menggunakan sentuhan jari.

Selama pengujian, aplikasi dapat menjalankan fungsi utama pada perangkat Android yang digunakan tanpa mengalami kendala yang menyebabkan aplikasi berhenti atau tidak dapat digunakan. Dengan demikian, perangkat Android yang digunakan dapat menjalankan aplikasi sesuai dengan kebutuhan rancang bangun yang telah ditetapkan.

Pembahasan

Hasil rancang bangun menunjukkan bahwa aplikasi *Augmented Reality* (AR) berbasis Android telah memenuhi fungsi utama sebagai media edukasi untuk mengenalkan komponen *hardware* komputer. Aplikasi memiliki beberapa halaman utama, yaitu halaman utama, *Scan Marker*, Info, Petunjuk, dan Informasi Pengembang. Melalui fitur *Scan Marker*, pengguna dapat memindai 10 *marker* yang mewakili *Monitor*, *Mouse*, *Keyboard*, *Casing CPU*, *Processor*, *Motherboard*, *Hard Disk Drive*, *Power Supply*, *Heatsink*, dan *RAM*. Setelah *marker* dikenali, aplikasi menampilkan objek tiga dimensi dan informasi sesuai dengan komponen yang dipindai. Pengguna juga dapat melakukan rotasi dan *zoom* pada objek 3D serta menggunakan tombol *Reset* untuk melakukan pemindaian terhadap *marker* lainnya. Penggunaan aplikasi oleh siswa SMP menunjukkan bahwa visualisasi objek 3D dapat membantu siswa mengenali bentuk komponen *hardware* komputer secara lebih interaktif.

Algoritma *Oriented FAST and Rotated BRIEF* (ORB) diterapkan untuk mendukung proses pendeteksian *marker* secara *real-time*. Proses tersebut meliputi pengambilan *frame* kamera, pemrosesan citra, ekstraksi *keypoints* dan *descriptor*,

pencocokan fitur menggunakan *Brute-Force Hamming*, serta penyaringan menggunakan *ratio test*. Hasil pencocokan kemudian divalidasi menggunakan *homography* dengan metode RANSAC, pemeriksaan *inlier*, bentuk dan luas area deteksi. Selanjutnya, *solvePnP()* digunakan untuk menentukan posisi dan orientasi *marker*. Hasil pengujian menunjukkan bahwa ORB dapat mengenali 10 *marker* yang telah didaftarkan dan menampilkan objek 3D yang sesuai. Namun, objek 3D belum dapat mengikuti perpindahan *marker* secara kontinu ketika *marker* digeser pada kamera. Objek masih berada pada posisi terakhir sampai pengguna melakukan *Reset* atau interaksi tertentu. Hal ini menunjukkan bahwa sistem telah mendukung pendeteksian *marker*, tetapi belum menerapkan *continuous tracking*.

Hasil pengujian aplikasi menunjukkan bahwa seluruh fungsi utama dapat berjalan sesuai dengan kebutuhan yang telah ditentukan. Pengujian *Black Box* terhadap 9 skenario memperoleh status *Pass* pada seluruh skenario dengan tingkat keberhasilan sebesar 100%. Pengujian deteksi *marker* juga menunjukkan bahwa aplikasi dapat mengenali *marker* komponen *hardware* dan menampilkan objek 3D yang sesuai. Selain itu, aplikasi dapat dijalankan pada perangkat Android yang digunakan dalam penelitian dan fitur utamanya dapat digunakan dengan baik. Hasil tersebut menunjukkan bahwa aplikasi telah berfungsi sesuai dengan rancangan dan dapat digunakan sebagai media edukasi pengenalan komponen *hardware* komputer.



Gambar 3 Penggunaan Aplikasi *Augmented Reality* oleh Siswa SMP

Secara keseluruhan, aplikasi AR berbasis Android yang dikembangkan telah memenuhi kebutuhan utama penelitian, yaitu menyediakan media edukasi yang memungkinkan siswa mengenali komponen *hardware* komputer melalui visualisasi objek 3D. Penerapan ORB mendukung proses pengenalan *marker*, sedangkan hasil pengujian menunjukkan bahwa fungsi aplikasi dapat berjalan dengan baik. Meskipun demikian, keterbatasan pada *tracking* objek 3D masih menjadi bagian yang perlu diperbaiki. Pengembangan selanjutnya dapat diarahkan pada penerapan pelacakan *marker* secara kontinu agar objek 3D dapat mengikuti perubahan posisi dan orientasi *marker* secara lebih stabil.

KESIMPULAN

Berdasarkan hasil rancang bangun, penerapan algoritma, dan pengujian yang telah dilakukan, dapat disimpulkan bahwa aplikasi *Augmented Reality* berbasis Android berhasil dirancang dan dibangun sebagai media edukasi pengenalan komponen *hardware* komputer bagi siswa SMP. Aplikasi menyediakan fitur pendeteksian *marker*, penampilan objek tiga dimensi, penyajian informasi komponen, serta interaksi terhadap objek melalui rotasi dan *zoom* menggunakan sentuhan jari. Aplikasi dapat digunakan untuk mengenalkan 10 komponen *hardware* komputer, yaitu *Monitor, Mouse, Keyboard, Casing CPU, Processor, Motherboard, Hard Disk Drive, Power Supply, Heatsink*, dan *RAM*, sehingga memberikan visualisasi yang membantu siswa dalam mengenali komponen komputer. Algoritma *Oriented FAST and Rotated BRIEF* (ORB) juga berhasil diterapkan dalam proses pendeteksian *marker* melalui tahapan pengambilan *frame* kamera, pemrosesan citra, ekstraksi *keypoints* dan *descriptor*, *feature matching*, validasi kecocokan menggunakan *homography* dan *RANSAC*, serta penentuan posisi *marker* menggunakan *solvePnP()*. Hasil pengujian menunjukkan bahwa sistem mampu mengenali 10 *marker* yang telah didaftarkan dan menampilkan objek tiga dimensi yang sesuai secara *real-time*, meskipun objek 3D yang ditampilkan belum mengikuti perpindahan *marker* secara kontinu dan tetap berada pada posisi terakhir hingga dilakukan *reset* atau interaksi terhadap objek. Selanjutnya, pengujian fungsional menggunakan metode *Black Box Testing* terhadap sembilan skenario menunjukkan bahwa seluruh fungsi utama aplikasi memperoleh status *Pass* dengan persentase keberhasilan sebesar 100%. Dengan demikian, fungsi pembukaan aplikasi, navigasi menu, pengaktifan kamera, pendeteksian *marker*, penampilan objek 3D, penyajian informasi *hardware*, dan keluar dari aplikasi telah berjalan sesuai dengan kebutuhan fungsional yang ditetapkan.

DAFTAR PUSTAKA

- Bjarnason, E., Lang, F., & Mjöberg, A. (2023). *An empirically based model of software prototyping: a mapping study and a multi-case study*. *Empirical Software Engineering*, 28(5), 115. <https://doi.org/10.1007/s10664-023-10331-w>
- Dinara, G., Zhumabekov, E., & Kadyrova, V. (2023). *Using OpenCV for real-time image recognition through augmented reality devices*. *Procedia Computer Science*, 219, 110–118. <https://doi.org/10.1016/j.procs.2023.01.015>
- Dinara, G., Zykov, E., & Kugurakova, V. (2023). *Using OpenCV for Augmented Reality Applications*. *Journal of Advanced Applied Learning Research*, 3(3), 179–184. https://doi.org/10.57417/jaalr.3.3_179
- Djafar, S., & Novian, D. (2021). *Implementasi teknologi augmented reality dalam pengembangan media pembelajaran perangkat keras komputer*. *Jambura Journal of Informatics*, 3(2), 85–94. <https://ejurnal.ung.ac.id/index.php/jji/article/view/10440>
- Google. (2025). *Meet Android Studio*. In *Android Developers*. Google. <https://developer.android.com/studio/intro>
- Gotoh, Y., & Ikeda, Y. (2025). *Augmented reality superimposed display system with*

- synchronized video for education. Multimedia Tools and Applications*, 84(3), 4121–4140.
<https://doi.org/10.1007/s11042-024-18523-9>
- Husár, J., Kaščák, J., & Hrehová, S. (2022). *Marker-based augmented reality in technical education. Education and Information Technologies*, 27(5), 6539–6558.
<https://doi.org/10.1007/s10639-021-10837-5>
- Imam, K., Yasin, V., & Sianipar, A. Z. (2021). Implementasi teknologi realitas augmentasi sebagai media pembelajaran interaktif. *Jurnal Widya*, 2(1), 45–53.
<https://www.jurnal.amikwidyaloka.ac.id>
- Kasan, T. F., & Suratmi. (2025). Pengembangan Media Pembelajaran 3D Augmented Reality Menggunakan Platform Sketchfab dan Assemblr Edu Materi Pencernaan Manusia. *Pedagogik: Jurnal Pendidikan Guru Sekolah Dasar*, 13(1), 162–172.
<https://doi.org/10.33558/pedagogik.v13i1.10827>
- Kementerian Pendidikan Riset, dan Teknologi, K. (2024). *Referensi Data Satuan Pendidikan: UPT SMP Negeri 7 Medan*. Kementerian Pendidikan, Kebudayaan, Riset, dan. <https://referensi.data.kemdikbud.go.id>
- Krishnamurthy, M., & Kabilamani, P. (2024). *Augmented reality applications in education: Challenges and opportunities. IEEE International Conference on Intelligent Systems*.
<https://doi.org/10.1109/ICIS56052.2024.11026596>
- Li, J., Liu, Y., Xiao, L., Cai, J., He, H., & Zhang, X. (2024). *Augmented reality creation platform using ORB feature detection. IEEE Access*, 12, 45678–45690.
<https://doi.org/10.1109/ACCESS.2024.3367891>
- Mazuera-Rozo, A., Escobar-Velásquez, C., Espitia-Acero, J., Vega-Guzmán, D., Trubiani, C., Linares-Vásquez, M., & Bavota, G. (2022). *Taxonomy of Security Weaknesses in Java and Kotlin Android Apps. Journal of Systems and Software*, 187, 111233. <https://doi.org/10.1016/j.jss.2022.111233>
- Santi, G. M., Ceruti, A., Liverani, A., & Osti, F. (2021). *Augmented reality in education: A systematic review. Technologies*, 9(2), 33.
<https://doi.org/10.3390/technologies9020033>
- Statistik, B. P. (2023). *Statistik Telekomunikasi Indonesia 2023*. Badan Pusat Statistik.
<https://www.bps.go.id>
- Sumardi, A. (2021). *Penerapan teknologi augmented reality menggunakan algoritma ORB pada aplikasi katalog rumah 3D [Universitas Komputer Indonesia]*.
<https://elibrary.unikom.ac.id/id/eprint/5601/>
- Umri, B. K., Wibowo, A. P. A., Widyananda, G. P. T., Nugraheni, A. I. D., & Hafizh, M. I. (2023). *Application of Augmented Reality Technology in Android Based Architectural Visualization. Jurnal Informatika Teknologi Dan Sains*, 5(4), 691–696.
<https://doi.org/10.51401/jinteks.v5i4.3458>
- Yuhano, & Akbar, F. (2024). *Development of Augmented Reality Application as An Educational Media for Visitors to Museum Pusaka Keraton Kasepuhan Cirebon Using Object Tracking Method and Fast Corner Detection Algorithm Based on Android. Jurnal Nasional Pendidikan Teknik Informatika: JANAPATI*, 13(1), 116–128.
<https://doi.org/10.23887/janapati.v13i1.68925>