

Yazılım Geliştirme Lab. I - Proje I

Binnur Özcan

Bilişim Sistemleri Mühendisliği, Teknoloji Fakültesi
Kocaeli Üniversitesi
binnur@outlook.com, 191307059@kocaeli.edu.tr

Özet

Bu proje, Android platformunda çalışan bir artırılmış gerçeklik (AR) uygulamasını içermektedir ve ARCore teknolojisini kullanmaktadır. Uygulama, `io.github.sceneview:arsceneview:0.9.8` bağımlılığı aracılığıyla ARCore'un sağladığı özellikleri kullanarak, SceneView kütüphanesi ve ARModelNode ile birlikte bir 3D modelin sanal dünyada yerleştirilmesini sağlamaktadır. Proje, Kotlin programlama dilinde geliştirilmiş olup, AR özellikleri ve kullanıcı etkileşimi için Android'in güncel kütüphanelerini kullanmaktadır.

Anahtar Kelimeler— Kotlin, Android, ARCore, ARModelNode

1. Giriş

Günümüzde teknolojinin hızla ilerlemesi, mobil uygulama geliştirme alanında çeşitli yenilikçi yaklaşımların ortaya çıkmasına olanak tanımaktadır. Bu bağlamda, bu proje kapsamında Android platformunda çalışan bir artırılmış gerçeklik (AR) uygulaması geliştirilmiştir. Artırılmış gerçeklik, fiziksel dünyayı dijital olarak zenginleştirerek kullanıcılara interaktif ve benzersiz deneyimler sunan bir teknolojidir.

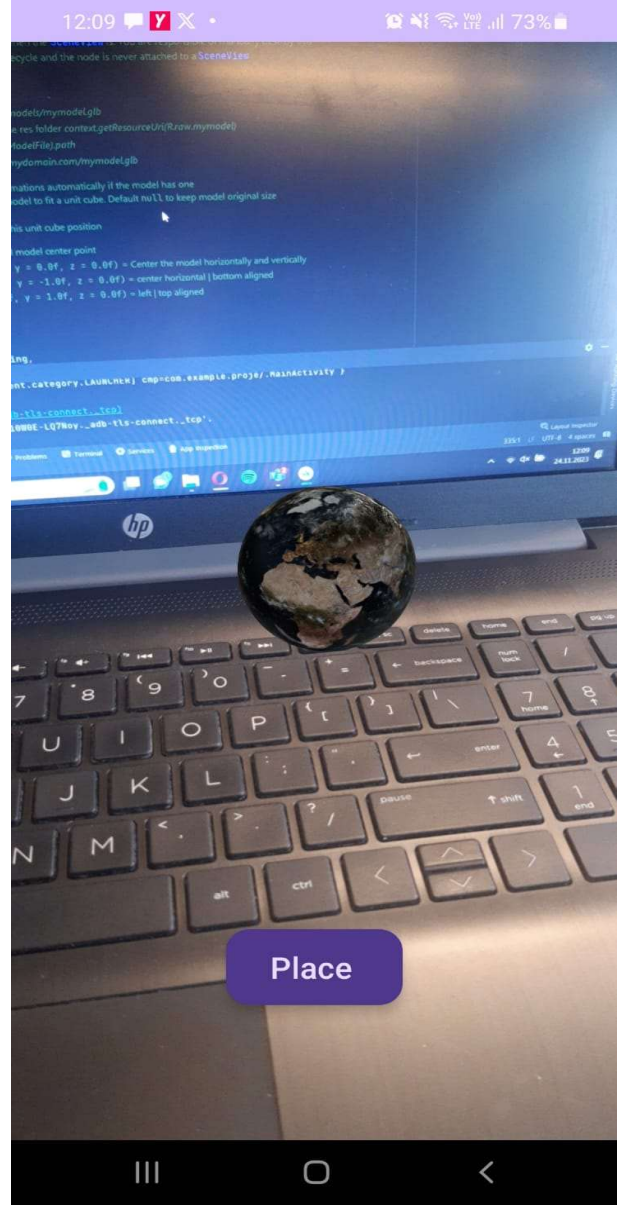
Bu projenin temel amacı, kullanıcıların mobil cihazları aracılığıyla gerçek dünyayı daha fazla bilgi ve içerikle zenginleştirmelerine olanak tanıyan bir AR deneyimi sunmaktır. Proje, Android ARCore API'sini temel alarak, SceneView kütüphanesi ve ARModelNode ile birlikte bir 3D modelin sanal dünyada yerleştirilmesini mümkün kılan bir altyapı sunmaktadır.

2. Uygulama Akışı

Uygulama başladığında, `onCreate` fonksiyonu çağrılır ve `activity_main.xml` dosyasındaki arayüz yüklenir.

Kullanıcı, ekranda "Place" düğmesine tıkladığında `placeModel` fonksiyonu çağrılır. `placeModel` fonksiyonu, `models/earth.glb` 3D modeli yükler ve AR sahnesine ekler. Kullanıcı, uygulama ekranını kameraya yönlendirerek çevresini görebilir. "Place" düğmesine tıkladığında, seçilen 3D model AR sahnesine yerleştirilir. Model hareket ettirilebilir.

Bu uygulama, temel olarak kullanıcılara belirli bir 3D modeli gerçek dünyada sanal olarak görmelerini sağlayarak artırılmış gerçeklik deneyimi sunar. Kullanıcılar, uygulama aracılığıyla seçtikleri 3D modeli kendi çevrelerine yerleştirebilir ve etkileşimde bulunabilirler.



3.

3.1. Kod Açıklamaları

Bu projenin `build.gradle` dosyası, Android uygulamasının temel yapılandırma ve bağımlılıklarını belirlemektedir.

- Plugins bloğu projenin Android uygulaması olarak tanımlanmasını ve Kotlin ile geliştirilmesini sağlayan eklentileri içerir.
- Android bloğu genel proje yapılandırmasıyla ilgili ayarları içerir. Özellikle, uygulamanın paket adı, derleme sürümü, yayın yapısı ve Kotlin derleme ayarları gibi temel konfigürasyonlar bu blokta yer alır.
- Dependencies bloğu projenin dış bağımlılıklarını belirtir. Özellikle, Android uygulaması için temel kütüphaneler (`core-ktx`, `appcompat`, `material`, `constraintlayout`) ve artırılmış gerçeklik özellikleri eklemek için kullanılan

arsceneview kütüphanesi gibi bağımlılıklar bu blokta tanımlanmıştır.

3.2. Activity_main.xml

Bu XML dosyası, Android uygulamanızın ana ekranını tanımlayan ve kullanıcının etkileşimde bulunabileceği bileşenleri içeren bir düzeni ifade eder.

Temel düzenini tanımlar ve bu düzen içindeki bileşenlerin yerleşimini belirler. ConstraintLayout kullanılarak bileşenler arasındaki ilişkiler ve konumlar belirlenmiştir. Bu düzen, kullanıcıya AR sahnelerini görüntüleme ve bir 3D modeli yerleştirme olanağı sağlayan temel bir arayüz sunar.

- <androidx.constraintlayout.widget.ConstraintLayout> Elemanı:

Bu eleman, ConstraintLayout adlı bir düzen yöneticisini temsil eder.layout_width ve layout_height özellikleri, elemanın ebeveynine (burada, match_parent ile tam ekran) göre genişliği ve yüksekliği belirler.

- <io.github.sceneview.ar.ArSceneView> Elemanı:

Bu eleman, AR sahnelerini görüntülemek için kullanılan ArSceneView sınıfını temsil eder.layout_width ve layout_height özellikleri, elemanın ebeveynine göre genişliği ve yüksekliği belirler.android:id özelliği, bu görünümü Java/Kotlin kodunda benzersiz bir şekilde tanımlamak için kullanılır.

- <com.google.android.material.floatingactionbutton.ExtendedFloatingActionButton> Elemanı:

Bu eleman, genişletilebilir bir kayan eylem düğmesini temsil eder.layout_width ve layout_height özellikleri, elemanın genişliği ve yüksekliği.android:id özelliği, bu düğmeyi Java/Kotlin kodunda tanımlamak için kullanılır.android:text, düğmenin üzerinde görünen metni belirtir (burada "Place").app:layout_constraint* özellikleri, elemanın ConstraintLayout içindeki konumunu belirler.

-

3.3. MainActivity.kt

Bu kod parçası, kullanıcının artırılmış gerçeklikle bir 3D modeli yerleştirmesini sağlayan temel işlevselliği içerir. ArSceneView, ArModelNode ve düğme etkileşimleri, kullanıcıya zengin bir AR deneyimi sunmak için entegre edilmiştir.

1. class MainActivity : AppCompatActivity() { ... }
Sınıf Tanımı:

Bu sınıf, AppCompatActivity sınıfından türetilmiş bir ana aktivite sınıfını temsil

eder.AppCompatActivity, modern Android uygulamalarının özelliklerini destekler.

2. Sınıf Üye Değişkenleri:

lateinit var sceneView: ArSceneView: AR sahnelerini görüntülemek için kullanılan ArSceneView sınıfından bir örnek. Geç başlatma (lateinit) kullanılarak ileride başlatılacaktır.lateinit var placeButton: ExtendedFloatingActionButton: Artırılmış gerçeklikle bir modelin yerleştirilmesini başlatan düğmenin örneği.private lateinit var modelNode: ArModelNode: AR sahnesine yerleştirilecek 3D modeli temsil eden ArModelNode sınıfından bir örnek.3. onCreate Fonksiyonu:

Bu fonksiyon, aktivite oluşturulduğunda çağrılır ve kullanıcı arayüzünü başlatır.

setContent(R.layout.activity_main): Kullanıcı arayüzünü activity_main.xml dosyasındaki düzenle eşleştirir.findViewById: XML dosyasındaki öğeleri bu sınıfa bağlar.placeButton.setOnClickListener { placeModel() }: "Place" düğmesine tıklandığında placeModel fonksiyonunu çağırır.

3. loadModelGlbAsync Bloğu:

ArModelNode() örneği oluşturulur ve içinde bir 3D model yüklenir.loadModelGlbAsync fonksiyonu, GLB dosyasının yüklenmesini sağlar. https://sketchfab.com adresinden alınan 3D model Android uygulamasına bu aşamada yüklenmiştir. Bu model sceneView.planeRenderer.isVisible = true: PlaneRenderer'ın görünürlüğü etkinleştirilir.onAnchorChanged: Bir nesne yerleştirildiğinde çalışacak bir işlev belirlenir.

4. placeModel Fonksiyonu:

Modelin yerleştirilmesini sağlar.modelNode?.anchor(): Model, AR sahnesine yerleştirilir.sceneView.planeRenderer.isVisible = false: PlaneRenderer'ın görünürlüğü devre dışı bırakılır.

3.4. Kaynakça

<https://sketchfab.com/>
<https://github.com/SceneView/sceneview-android/blob/main/README.md>
<https://github.com/princeku07/AR-Furniture>
<https://medium.com/@nivethsaran/build-a-simple-android-ar-application-in-10-steps-6a63571e2e5d>
<https://thinkupsoft.com/blog/sceneform-making-arcore-easier/>