

## 3B Yapı Modellerinin Sanal Küre Üzerinde Gösterimi

İlhan YILDIRIM<sup>1\*</sup>, Burhan SELÇUK<sup>2</sup>

<sup>1</sup>Harita Genel Müdürlüğü / Milli Savunma Bakanlığı, Ankara, Türkiye

<sup>2</sup>Bilgisayar Mühendisliği Bölümü / Mühendislik Fakültesi, Karabük Üniversitesi, Türkiye

\*ilhanyildirim27@gmail.com

**Özet** – Coğrafi mekânsal verinin doğasında olan karmaşa ve düzensizlik, kullanıcılar tarafından doğru algılanmasını zorlaştırmakta ve görselleştirilmesinin önemini arttırmaktadır. 3B, gösterimler, mekânsal nesnelerin algılanmasını kolaylaştırmaktadır. 3B kent modelleri farklı amaçlar için birçok disiplin tarafından kullanılmakta ve bu alandaki ihtiyaçlar gün geçtikçe artmaktadır. OpenGL teknolojisinin geliştirilmesi ve WebGL konseptinin tanıtılmasıyla web ortamına ve masaüstü tarayıcılarına iletilmesi, 3B animasyonların son kullanıcılara aktarılması sürecini büyük ölçüde kolaylaştırmıştır. Çevrimiçi 3B sanal küreler, 3B şehir verilerini tarayıcılara aktarmak için WebGL teknolojisinden yararlanmaktadır ve aktarılan veri yığınlarını kullanıcının kişisel bilgisayarları ve akıllı telefonların donanımından yararlanarak işleyip görselleştirmektedir. Bu kısaca 3B sanalın genel mimarisini açıklar. WebGL desteğinin gelmesiyle beraber 3B şehir modellerinin, masaüstü uygulamaların tarayıcılara taşınması kaçınılmaz olmuştur. Şehrin yapısını ve binaların haritadaki yerlerini gösteren verilerin olması güvenlik güçlerinin yapacağı operasyonlar için büyük önem arz etmektedir. Bu yüzden operasyon bölgelerinde bulunan yapıların 3B halde harita üzerinde gösterimine ihtiyaç duyulmuştur. JavaScript kütüphanesi olan Angular uygulama geliştirme ve yine JavaScript kütüphanesi olan Cesium sanal küresi kullanılarak, 3B şehir modelleri küre üzerinde gösterilmiştir. 3B şehir modellerinin hazırlanması içinde FME Workbench, veri dönüştürme programı kullanılmıştır. Bu programla veriler Cesium sanal küresinin desteklediği “b3dm” formatına çevrilmiştir

*Anahtar Kelimeler – 3B Şehir Modelleme, 3B Görselleştirme, Cesium, Sanal Küre, WebGL*

### I. GİRİŞ

Bilgisayar ve ağ teknolojisinin gelişmesiyle birlikte, WebGL teknolojisi ortaya çıkmıştır. WebGL, kâr amacı gütmeyen teknoloji şirketleri birliği Khronos Group tarafından yönetilen, web ortamında 3B grafik çizimi sağlanmasını amaçlayan bir web standardıdır. HTML 5 ile çıkan bu sitem, internet tarayıcıları üzerinde 3B grafik oluşturmak için kullanılan platform bağımsız ve ücretsiz bir programlama ara yüzüdür. WebGL’in teknolojisinin gelişmesiyle beraber web üzerinde 3B modelleme ve Coğrafi Bilgi Sisteminin (CBS) entegrasyonu bir ihtiyaç haline gelmiştir. Günümüzde 3B modelleme, birçok alanda kullanılmaktadır. Bunlardan en önemlileri; mimari, mühendislik, inşaat, eğitim ve askeri alanlardır.

CBS ile 3B modellerin entegrasyonu, etkin bir şekilde teşvik edilebilir ve akıllı şehirlerin inşasını gerçekleştirebilir. Fakat ağ bant genişliği ve sunucu performansındaki sınırlamalar nedeniyle, 3B modellerin yüklenmesinde, genellikle uzun yükleme süresi ve sistem çökmeleri gibi sorunlarla karşılaşmaktadır. Birçok çalışma bu sorunu 3B model üzerinde basitleştirmeler ile ele almıştır.

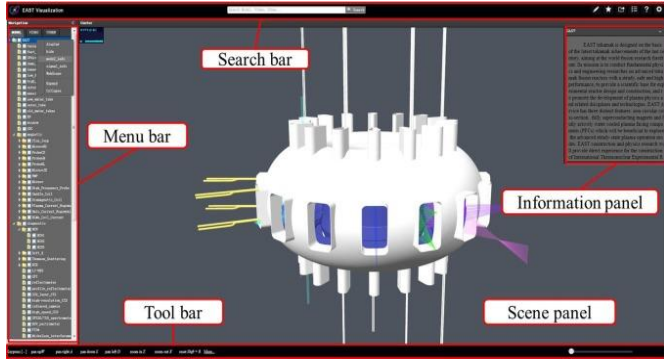
3B model içerisinde birçok özelliği barındırabilir; bu nedenle, sorun yaşanan model üzerinde basitleştirme, çok büyük miktarda bilginin kaybına neden olabilmektedir. Bu nedenle, büyük 3B modellerin Web’e yüklenmesi sorunu şuan için çözülmeyen problemler sınıfındadır. 3B modelleri sunmak için standart olarak WebGL’i de kullanılan Cesium son yıllarda yaygın olarak kullanılmaktadır. Cesium, bir web tarayıcısında 3B sanal küre ve iki

boyutlu düz haritalar sunmak için kullanılan açık kaynaklı bir JavaScript kütüphanesidir.

Bu çalışma, 3B binaların WebGIS ortamına hızlı bir şekilde yüklenebilmesini sağlayan bir yöntemi ele almıştır. Bu nedenle bu çalışma, binaların 3B veri formatına dönüştürmeyi ve bu binaları sanal bir küre üzerinde sunmayı amaçlamaktadır.

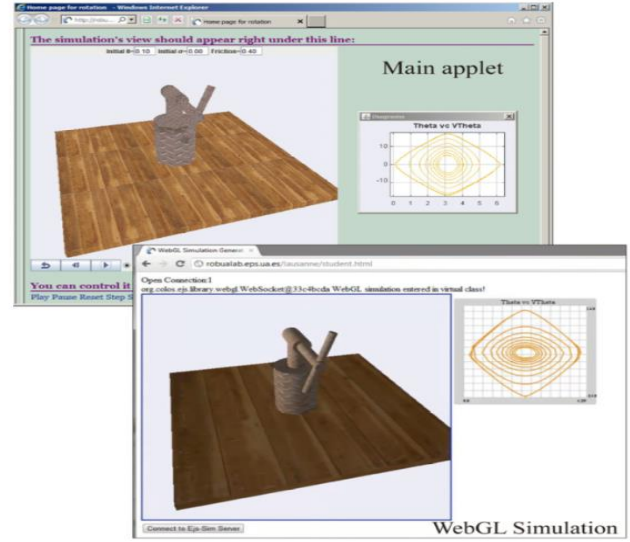
## II. LİTERATÜR ÖZETİ

Son yıllarda EAST (manyetik füzyon enerjisi, güneşteki nükleer füzyon sericini taklit ederek sabit ve temiz enerji üretmek) deneysel verileri, çok sayıda uluslararası işbirlikçi tarafından paylaşılmakta ve analiz edilmektedir. Verileri, meta verileri ve cihazın ilgili parçalarını görselleştirmek için kullanıcı dostu bir ara yüz geliştirmek, bilim adamlarına ve mühendislere yardımcı olmak için giderek daha önemli hale geliyor. Xia ve arkadaşları EAST deneyi için birbiri ile etkileşimli WebGL tabanlı 3B görselleştirme sistemini önermişlerdir. Önerdikleri sistemde kullanıcılar sanal sahnede dolaşarak, EAST'in iç yapısını görebilir, bileşenlere tıklayarak konum parametresini ve ilgili deneysel verilerini alabiliyorlar. (Xia ve ark. 2016)



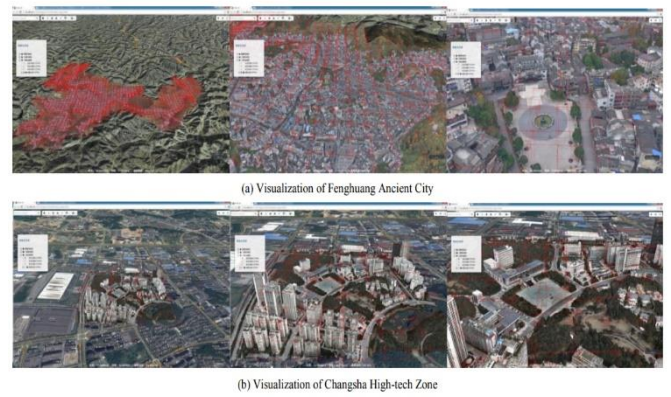
Şekil 1. EAST 3B sistemin kullanıcı arayüzü

Carlos ve arkadaşları sanal laboratuvarlar arasında senkronize edilmiş, gerçek zamanlı ve iletişime dayalı yeni bir e-öğrenme sistemi geliştirmişlerdir. Bu sistem, 3B Simülasyon uygulamalarından WebGL simülasyonlarının otomatik olarak oluşturulmasına dayanmaktadır. Oluşturulan bu WebGL simülasyonları, öğrencileri internet üzerinde çevrimiçi bir ortama entegre ederek senkronize ediyor. Bu şekilde bir öğrenci, WebGL'in etkin olduğu herhangi bir web tarayıcısını kullanarak yalnızca bir iletişim cihazı kullanarak sanal bir sınıfa katılabilir. (Jara ve ark. 2012)



Şekil 2. Sanal Laboratuvar Ara yüzü

3B bina modelleri büyük veri boyutuna sahip olabilirler. Bina modelleri farklı tarayıcı ve farklı cihazlarda çalışabilir olmalıdırlar. Bunun için Song ve arkadaşları terminal bellek boyutuna ve ekran alanı hatasına dayalı olarak bir 3B döşeme, dinamik yükleme ve zamanlama stratejisi önermişlerdir. Bu strateji, görsel etkileşim süreci sırasında gerçek zamanlı olarak veri yükünü izler. Yüklenen veriler önceden ayarlanmış eşiği aştığında, yerine daha düşük çözünürlüklü döşemeler kullanılarak bazı yüksek çözünürlüklü döşemeler kaldırılır. Böylelikle görsel etkileşim ve veri yüklenmesi sırasında daha iyi görsel etkiler sağlanır ve tarayıcının her zaman normal şekilde çalışmasını sağlanmaktadır. (Song ve ark. 2018)

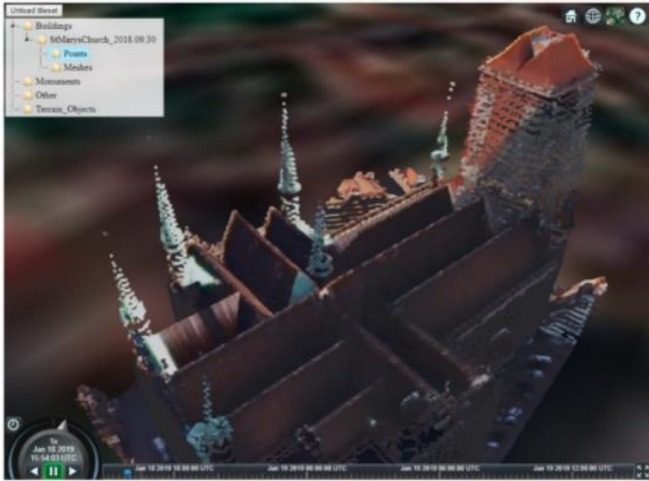


Şekil 3. Eşik Seviyesine Göre 3B Binaların Görüntüsü

3B lazer tarama tekniğiyle çalışan LIDAR 1960'lı yıllarda havadan deniz altlarının tespiti için geliştirilmiş olup, 1970'li yıllarda kullanılmaya başlanmıştır. Daha sonraki yıllarda ise hem havadan hem de karadan olmak üzere LIDAR tekniği

kullanan sensörlerin kullanım alanı ve çeşitlerinde artış olmuştur. LIDAR teknolojisi günümüzde mimarlık, arkeoloji, şehir planlaması, petrol ve doğal gaz aramaları, haritalandırma, orman ve su araştırmaları alanlarında kullanılmaktadır. Kulawiak ve arkadaşları, bir web ortamında LIDAR verilerinin 3B modellerinin entegrasyonu, işlenmesi ve görselleştirilmesi için verimli bir uygulama geliştirmişlerdir. Bu uygulama aşağıdakiler dahil olmak üzere mevcut bireysel sistemlerin yeteneklerini sunmaktadır;

- Farklı kullanıcılar tarafından sağlanan verilerin yüklenmesi, entegrasyonu ve paylaşımı,
- Binaların nokta bulutlarından 3B model yapımı,
- Açık kaynaklı formatlar ve protokoller kullanılarak verilerin depolanması ve dağıtılması (Kulawiak ve ark. 2019)

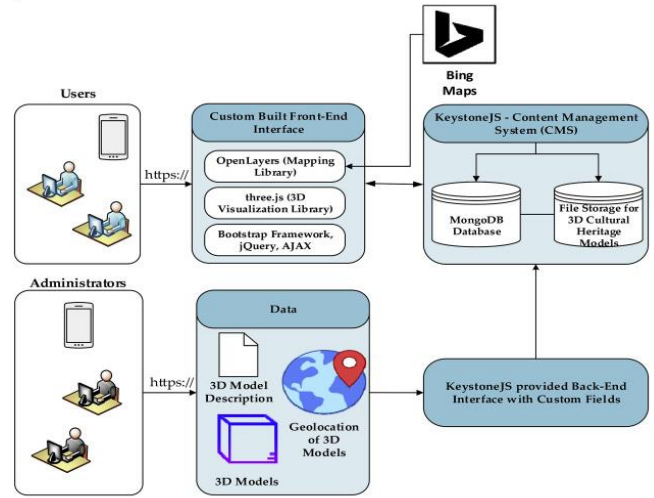


Şekil 4. LIDAR Verilerinin 3B Gösterimi

Son yıllarda Çin’de tarım ürünlerinin üretiminde ciddi yükselişler gözlemlenmektedir. Fakat üretimin yanında bazı bölgelerde ciddi durumda doğal afetler meydana gelmektedir. Buda üreticinin doğru ve güvenilir bir yerde üretim yapma arayışına yönlendirmiştir. Yufeng ve Yuzhe, güvenli ve az riskli üretim için 3B görseller geliştirmişlerdir. ArcGIS ve FME aracılığıyla 2B verileri 3B’ya dönüştürmüşlerdir. ArcGIS Servis platformu ile güçlü bir veri dönüştürme aracı olarak FME, eksiksiz ve etkili bir sunucu tarafı verileri ile birlikte çalışabilirlik mimarisi sağlar. (Yufeng and Yuzhe 2013)



Şekil 5. Güvenli Üretim Bölgeleri



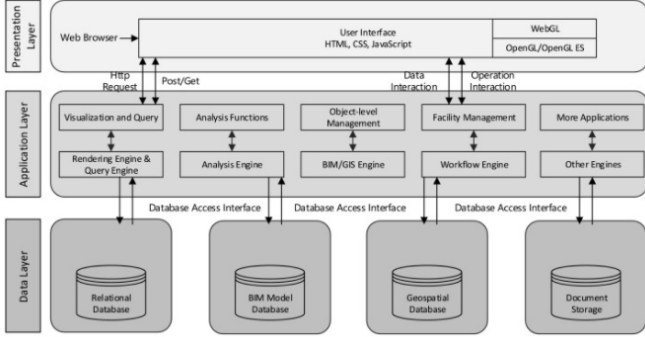
Şekil 6. 3B Modellerin Oluşturulması ve Depolanması



Şekil 7. 3B Dijital Kültürel Miraslar

Hidrolik ve hidroelektrik mühendisliği projeleri, büyük ölçekli, uzun inşaat süresi, karmaşık teknoloji, karmaşık yöntem ve yoğun bir koordinasyon gerektirmektedir. Zhang ve ark. Hidrolik ve hidroelektrik projeleri için 3B model ile Coğrafi Bilgi Sistemlerini (CBS) entegre edilmiş bir entegrasyon platformu (BGIP) geliştirmişleridir. Bu platform, 3B model bilgisi ve coğrafi bilgisini birleştirmek için modelin depolama, doğruluk ve

çalışma hızındaki tutarsızlıkları çözmek için tatmin edici bir şema önermişlerdir. Bu platformda üç ana konuyu ele almışlardır; geometrik görselleştirme ve koordinat dönüşümü, alan odaklı bilgi sorgulama ve oluşturmadır. (Zhang ve ark. 2020)



Şekil 8. Sistem Mimarisi

Birinci katman sunum katmanıdır. İkinci katman uygulama katmanıdır. Üçüncü katman ise veri katmanıdır.



Şekil 9. 3B Modelin Görüntüsü

### III. MATERYAL VE YÖNTEM

Bu çalışmada, 2B binalar FME programı sayesinde 3B binalara dönüştürülmüştür. Daha sonra Javascript kütüphanesi olan Angular ile bir proje oluşturulmuştur. Bu projeye açık kaynaklı olan, 3B verileri ve 2B haritaları sunmamızı sağlayan Cesium kütüphanesi eklenmiştir. FME programı ile dönüştürdüğümüz 3B binaları Cesium kütüphanesi sayesinde hem 3B harita üzerinde hem de gerçek konumunda kullanıcılara sunulmuştur.

Burada kullanılan harita althığı için ücretsiz ve herkese açık olan Microsoft virtualearth kullanılmıştır. Sanal küre üzerindeki yükseklik verileri de Cesium kendisinden sağlanmıştır.

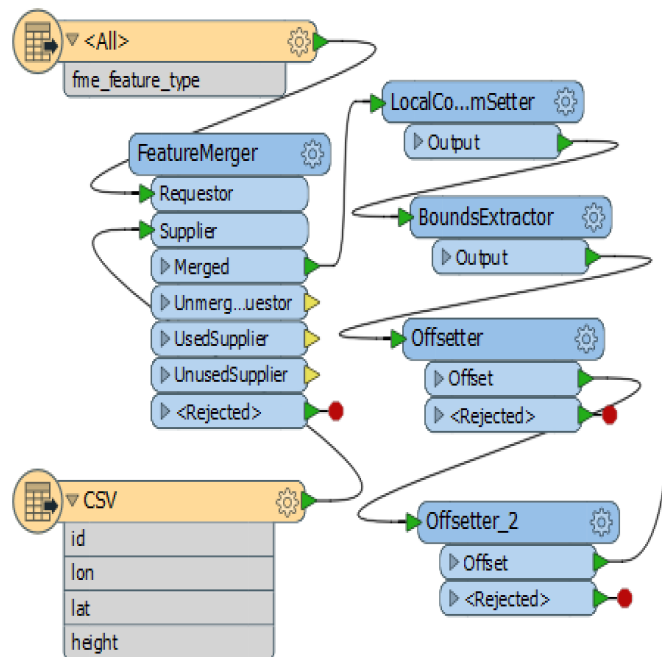
İşlemler ASUS FX505GD-BQ13916 Intel Core i7 8750H, 16GB RAM, 256GB SSD, 2.2 Ghz ve 8GB ekran kartı özelliklerine sahip dizüstü bilgisayarda gerçekleştirilmiştir. Programlama dili olarak Javascript, geliştirme ortamı olarak Visual Studio Code ve Node.js kullanılmıştır.

#### A. FME Workbench

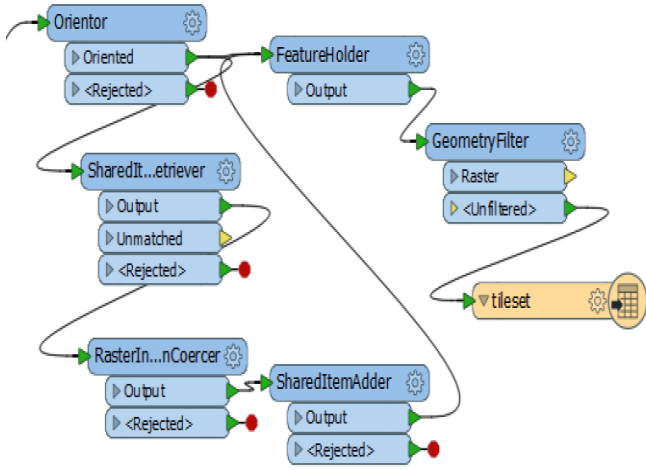
FME (Özellik Manipülasyon Mühendisliği), Safe Software Inc. tarafından üretilen veri dönüşümü ve veri çevirisi için ETL (Ayıkla, Dönüştür, Yükle) araçlarının entegre edilmiş bir koleksiyonudur. Verileri yeniden yapılandırarak gerçekleştirilen anlamsal dönüşüme dayanır, farklı türdeki son kullanıcı veya sistem taleplerini karşılamak için kaynak verilerini farklı düzeylerini ve içeriğini çıkarmak için anlamsal eşleme dosyasını kullanır. Bir genişletme modülü olarak FME, veri birlikte çalışabilirliği olarak adlandırılan ArcGIS'e gömülüdür. (Yufeng and Yuzhe 2013)

#### B. Dönüşüm Süreci

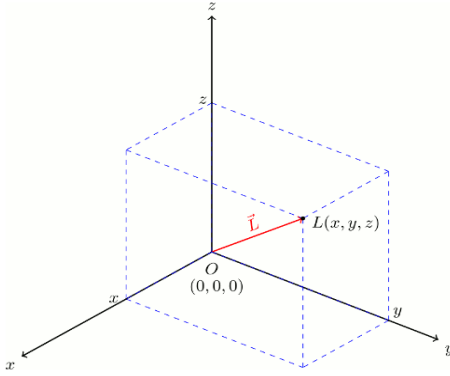
3D Tiles verilerini doğrudan elde etmek mümkün değildir. Elimizde binalara ait OBJ uzantılı dosyalar, koordinatlar ve binaların denk geleceği konumun yükseklik bilgisi olan MAP dosyasının olması gerekiyor. OBJ dosyaları geometrik bilgileri ve malzeme özelliklerini içerir. Bu nedenle bir 3B bina modelini dönüştürmek için mükemmel bir ortam görevi görür. MAP dosyasındaki yükseklik verisi eklenmez ise 3B bir haritada binalar havada kalır. FME programına bu parametreler verildikten sonra b3dm ve tileset.json dosyaları oluşur. Binalar hakkındaki bilgiler JSON dosyasında tutulur. OBJ dosyaları b3dm verilerine gömülür. Sonuç olarak 3D Tiles verileri b3dm verilerinden ve tileset.json dosyalarından oluşur.



Şekil 10. FME, 3B Bina Üretim Şeması -1



Şekil 11. FME, 3B Bina Üretim Şeması -2



Şekil 12. 3B, Kartezyen Koordinat Sistemi

### C. Node.js

Ryan Dahl tarafından 2009 yılında Google Chrome tarayıcısının JavaScript komutlarını çalıştırmak için kullandığı V8 JavaScript motoruna çeşitli eklemeler yaparak JavaScript komutlarının sunucu tarafında çalışması için geliştirilmiştir.

V8 motoru C/C++ ile geliştirilmiş JavaScript komutlarını makine diline çevirmek için kullanılan bir ara yazılımdır.

Komutların makine koduna çevrilmesi JavaScript komutlarının daha hızlı ve performanslı çalışmasını sağlar.

### D. Angular

Angular, dinamik web uygulamaları için kullanılan yapısal bir framework'tür. HTML şablon dili olarak kullanmayı sağlamakla birlikte en büyük avantajlarından biri ise Google destekli olmasıdır.

Bu sebepten dolayı yaygın olarak kullanılmaktadır. Diğer JavaScript kütüphanelerine göre daha kısa yazılan kodlar sayesinde çok daha fazla iş yapılabilmesini sağlamaktadır.

Angular ile yığın kod yazmak zorunda kalmadan kodun çoğunu ortadan kaldırarak neredeyse çoğu tarayıcıda bulunan ve herhangi bir sunucu teknolojisi ile ideal bir ortaklık yaparak çalışmayı sağlar. JQuery gibi kütüphanelere ihtiyaç duymaz. Asıl amacı tek bir sayfa üzerinden dinamik olarak geliştirme olanağı sağlar. Benzer yapıları kütüphanelerden en büyük farkı çift yönlü olmasıdır. Yani veride olan bir değişiklik görünümü, görünümde olan değişiklik veriyi değiştirir. Angular uygulamaları JavaScript'in üzerine inşa edilmiş bir dil olan TypeScript'in Html, Css ile birlikte kullanılmasını içerir. TypeScript'te yazılan kod önce JavaScript'e derlenir ardından tarayıcıda işlenir.

İlk başta Angular CLI kurulur. Angular CLI, Angular uygulamaları için oluşturulmuş EmberCLI temelli bir yapıdır. CLI'in açılımı, Command Line Interface (Komut Satırı Arayüzü) şeklindedir. Angular uygulamalarında kullanılacak projeler, klasörler, dosyalar ve test gibi birçok işlemi hızlı bir şekilde yapmamızı sağlayan araçtır. Angular CLI kurulumu için terminal ekranında "npm install -g @angular/cli" komutu girilir ve enter'a basılır. CLI kurulumu tamamlandıktan sonra yeni proje oluşturulur. Bunun için "ng new ProjeAdı" komutu yazılarak enter'a basılır. Proje oluşturduktan sonra "ng serve --open" komutu yazılarak projemiz çalıştırılır. (Eygi 2020)

### E. 3D Tiles ve Cesium

WebGL standardını temel alan Cesium, bir web tarayıcısında 3B sanal küreler ve 2B ile 3B haritalar oluşturmak için açık kaynaklı bir JavaScript kütüphanesidir. Cesium, konumsal verilere tutarlı bir şekilde erişir ve hem kamu hem de özel veri kaynaklarını destekler. Cesium, OGC'nin WMS, WFS ve diğer ağ hizmeti özelliklerini destekler. Uzak servisler aracılığıyla sunucu tarafındaki harita verilerini yükleyebilir ve ardından harita verilerini tarayıcıda görsel olarak sunabilir. 3D Tiles, verimli bir şekilde yüklenebilen bir veri formatıdır. Cesium, 3D Tiles formatını destekler.

## F. Cesium Kütüphanesinin Entegre Edilmesi

Bir Angular uygulamasına Cesium eklemek için önce npm'den yüklememiz gerekiyor. Bunun için projeyi Visual Studio Code'dan açtıktan sonra komut satırında “npm install –save cesium” komutunu yazıp enter'a basıyoruz. Cesium yüklendikten sonra projemizdeki angular.json dosyasına CesiumJS, stiller ve Css ayarları Şekil 12'deki gibi yapılır.

```
"assets": [
  {
    "glob": "**/*",
    "input": "../node_modules/cesium/Build/Cesium",
    "output": "../assets/cesium"
  },
  "assets",
  "favicon.ico"
],
"styles": [
  "../node_modules/cesium/Build/Cesium/widgets/widgets.css",
  "styles.css"
],
"scripts": ["../node_modules/cesium/Build/Cesium/Cesium.js"]
```

Şekil 13. Cesium Ayarlarının Yapılması

Uygulamamız sadelik adına sadece bir modül ve component'den oluşmaktadır. Component, Cesium küreyi barındırmak için kullanılacaktır. Component'e görüntüleyici eklemek için directive oluşturmamız gerekiyor. Bunun için komut satırından uygulama dizinine gidip “ng generate directive cesium” komutu çalıştırılır. Directive, Cesium küre nesnesini oluşturmak ve başlatmak için kullanılacaktır. Angular belgelerine göre, bir directive'deki veya component'deki başlatma mantığı, OnInit arabiriminin bir parçası olan ngOnInit işlevinin içine yerleştirilir. Cesium.directive.ts içerisine Şekil 13'deki komutlar yazılır.

```
import { Directive, OnInit } from '@angular/core';

@Directive({
  selector: '[appCesium]'
})
export class CesiumDirective implements OnInit {
  constructor(private el: ElementRef) {}

  ngOnInit() {
    // Put initialization code for the Cesium viewer here
  }
}
```

Şekil 14. cesium.directive.ts içine yazılacak kodlar

Directive'in eklendiği HTML ögesine bir referans almak için ngOnInit içine ElementRef ( **const viewer = new Cesium.Viewer (this.el.nativeElement)**) sınıfını ekleriz. Cesium görüntüleyici bu referansla başlatılır.

Daha sonra app.component.html dosyası “<div appCesium></div>” kodu ile değiştirilir. Style.css dosyasına küre için Şekil 14'deki stiller eklenir.

```
html, body, div[appCesium] {
  width: 100%; height: 100%; margin: 0; padding: 0; overflow: hidden;
}
```

Şekil 15. Style.css dosyasına eklenecek stiller

Son olarak projemizdeki main.ts dosyasına, Cesium'a statik dosyaları yüklemek için temel URL'yi söyleyen bir ortam değişkenini Şekil 15'deki gibi tanımlıyoruz.

```
window['CESIUM_BASE_URL'] = '/assets/cesium/';
```

Şekil 16. Main.ts dosyasına eklenecek URL kodu.

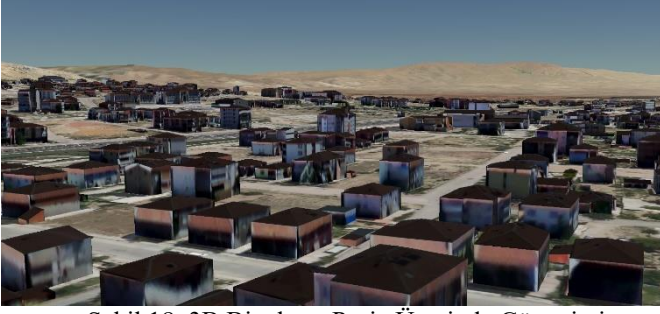
Eğer yüklediğimiz Cesium versiyonu 1.4.2.0 veya bundan yüksek ise URL'i “Cesium.buildModuleUrl.setBaseUrl('/assets/cesium/');” şeklinde tanımlıyoruz. Projemize Cesium'u eklemiş oluyoruz. Daha sonra “ng serve” komutuyla projemizi çalıştırıp Cesium küremizi görmüş oluruz. (Bampakos 2018)

## G. Uygulama

FME programı ile b3dm formatına dönüştürdüğümüz 3B binaları, Angular projemize eklememiz gerekiyor. Bunun için cesium.directive.ts dosyasının içine Şekil 3.5'deki kodları yazmamız gerekiyor. Burada URL kısmına FME çıktısı olan 3B binalarımızın dosya yolunu tanımlıyoruz. 3B binalarımızın yani b3dm uzantılı dosyalarımızın bulunduğu data klasörü ile tileset.json dosyası aynı dizinin altında olması gerekiyor.

```
add3DBuildings() {
  var d3Layer = this.viewer.scene.primitives.add(
    new Cesium.Cesium3DTileset({
      url: '../assets/hakkari/tileset.json',
    })
  );
  d3Layer.show = true;
  this.viewer.zoomTo(d3Layer);
}
```

Şekil 17. 3B binaların Cesium Küresine Entegre edilmesi



Şekil 18. 3B Binaların Proje Üzerinde Gösterimi

#### IV. SONUÇLAR VE ÖNERİLER

FME workbench programı OBJ dosyalarının geometrik bilgilerini çıkarabilir, format dönüştürme için öznitelik bilgilerini alabilir ve ilgili 3D Tiles dosyalarını alabilir. Format dönüştürme sırasında, bina verilerinin sanal indeks yapısı tasarlanır, bileşenin koordinat sistemi dönüştürülür. 3B'lu haritalarda binaların yerin altında kalmaması veya havada kalmaması için yükseklik bilgisi FME programına parametre olarak verilip istenilen yükseklikte 3B'lu model oluşturulabilmektedir.

Harita üzerine 3B model ekleme ve bu modellerin koordinatlarına denk gelecek şekilde yerleştirmek mümkündür. Performans ve sonuçlar, önerilen yaklaşımın daha geniş bir alanda ve daha farklı amaçlar için kullanılabileceğini göstermektedir. 3D Tiles, kararlılık ve verimlilik açısından diğer veri biçimlerine göre avantajlara sahip olsa da elde edilen şekil dosyasının daha geniş kullanımı ile daha fazla geliştirme gereklidir. Bu çalışmanın özelinde Aşağıdakiler de dahil olmak üzere gelecekteki gelişmeler beklenmektedir;

Bu çalışmada sunulan 3B bina verilerinin doğruluğu iyileştirilmelidir. 3B bina ile altlık olarak kullanılan harita örtüşmemektedir. 3B bina bazen haritadaki alandan büyük veya küçük olabiliyor.

3B bina dosyalarını düzenlemek zorlu bir görevdir. 3B binaların düzenlenmesinde binalar tek tek çizilmektedir ve çok zaman kaybına neden olmaktadır. 3B binaların düzenlenmesi otomatik hale getirilmelidir.

Düşük çözünürlüklü bilgisayarlarda 3B binaların yüklenmesi çok zaman almaktadır. Daha düşük boyutlu ve daha kaliteli bina modelleri oluşturulmalıdır.

#### KAYNAKLAR

[1] Bampakos, A. (2018). "Cesium and Angular." [Cesium](#).

- [2] Eygi, C. (2020). "Angular 9 for Beginners — How to Install Your First App with Angular CLI." [Freecodecamp](#).
- [3] Jara ve ark., C. A. (2012). "Synchronous collaboration between auto-generated WebGL applications and 3D virtual laboratories created with Easy Java Simulations." [IFAC Proceedings Volumes](#) **45**(11): 160-165.
- [4] Kulawiak ve ark., M. (2019). "Integration, processing and dissemination of LiDAR data in a 3D web-GIS." [ISPRS International Journal of Geo-Information](#) **8**(3): 144.
- [5] Nishanbaev, I. (2020). "A web repository for geo-located 3D digital cultural heritage models." [Digital Applications in Archaeology and Cultural Heritage](#) **16**: e00139.
- [6] Song ve ark., Z. (2018). [A dynamic tiles loading and scheduling strategy for massive oblique photogrammetry models](#). 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), IEEE.
- [7] Xia ve ark., J. (2016). "Interactive WebGL-based 3D visualizations for EAST experiment." [Fusion Engineering and Design](#) **112**: 946-951.
- [8] Yufeng, Z. and D. Yuzhe (2013). 3D model building in safe production based on FME. 2013 IEEE 4th International Conference on Electronics Information and Emergency Communication, IEEE.
- [9] Zhang ve ark., S. (2020). "Integrating and managing BIM in 3D web-based GIS for hydraulic and hydropower engineering projects." [Automation in Construction](#) **112**: 103114.