

Deprem sonrasında nokta bulutlarından hasarlı binaların belirlenmesine yönelik yapay öğrenme modeli tasarımı: ‘EQDamage4B_Pnt’

Caner Güney^{1*} 

¹İstanbul Teknik Üniversitesi, Maslak Kampüsü, İnşaat Fakültesi, Harita Mühendisliği Bölümü, Sarıyer, İstanbul, Türkiye.

Öz: Deprem sonrasında afet bölgesine ilişkin havadan elde edilmiş üç boyutlu (3B) nokta bulutlarından mevcut durumun 3B kent modelinin yani 3B sahnenin üretilmesi ve bunun üzerinde 3B bölütleme gibi bilgisayarlı görü görevlerinin gerçekleştirilmesi önemli bir konudur. Buradan hareketle deprem sonrasında elde edilen büyük ve karmaşık nokta bulutlarının 3B bölütlenmesi görevinde hem yapay öğrenme modelinin temsil gücünü hem de hesaplama verimliliğini arttıracak bir mimarinin tasarlanması ve bu mimariye dayalı bir modelin geliştirilmesi kritik öneme sahiptir. Ayrıca yapay öğrenme alanında da kullanılabilecek en iyi mimariden ve modelden ziyade, projenin amacına ve gereksinimlerine en uygun mimariyi seçebilmek ve buna ilişkin modeli geliştirebilmek çok daha önemli görülmektedir. Yapay öğrenme yaklaşımında; araştırma ve uygulama gereksinimlerine göre modelin ölçeklenebilirliği, ön eğitim yaklaşımları ve bütünleştirme kolaylığı gibi ölçütler mimari ve model seçiminde belirleyici olmaktadır. Çalışma kapsamında deprem sonrasında düzensiz ve sırasız yapıda bulunan nokta bulutlarından bina hasar tespiti için yüksek detay ve çok ölçekli bilgi gereksinimleri göz önüne alındığında SONATA öz denetim ön eğiticisinin, PointTransformer v3 özellik çıkarıcısının, Mask3D bölütleme başlığının, PointLLM etkileşiminin, Point2Surf grafik modelleyicisinin bir süreç tasarımı geliştirilmiş ve EQDamage4B_Pnt olarak adlandırılmıştır. Süreç tasarımı sonucunda elde edilen mimari ile bina hasar tespiti, deprem sonrası hızlı hasar değerlendirmesi ve müdahale planlaması gibi görevlere yönelik çözümler uçtan-uca yapay öğrenme tabanlı olarak üretilebilir yapıya getirilmiş olacaktır.

Anahtar Sözcükler: Deprem, Hasar belirleme, Nokta bulutu, LiDAR, Segmentasyon, Derin öğrenme

Design of a machine learning model for identifying damaged buildings from post-earthquake point clouds: ‘EQDamage4B_Pnt’

Abstract: After an earthquake, it is important to produce a 3D city model, i.e., a 3D scene, from 3D point clouds obtained from the sky regarding the disaster area and to perform computer vision tasks such as 3D segmentation on it. Based on this, designing an architecture that enhances both the representational power of the artificial intelligence model and computational efficiency, and developing a model based on this architecture, is of critical importance in the task of 3D segmentation of the large and complex point clouds obtained after the earthquake. Furthermore, rather than selecting the best architecture and model available in the field of artificial intelligence, it is considered much more important to choose the architecture that best suits the project's objectives and requirements and to develop the corresponding model. In the machine learning approach, criteria such as model scalability, pre-training approaches, and ease of integration are decisive in architectural and model selection, depending on research and application requirements. Considering the high-detail and multi-scale information requirements for building damage detection from point clouds with irregular and disordered structures after an earthquake, a process design combining the SONATA self-supervised pre-trainer, PointTransformer v3 feature extractor, Mask3D segmentation head, PointLLM interaction, and Point2Surf graphical modeler have been combined to develop a process design named EQDamage4B_Pnt. The resulting architecture enables end-to-end machine learning-based solutions for tasks such as building damage detection, post-earthquake rapid damage assessment, and intervention planning.

Keywords: Earthquake, Building damage assessment, Point cloud, LiDAR, Segmentation, Deep learning

* Sorumlu Yazar/Corresponding Author: Tel: +90 212 285 3825

Geliş Tarihi/Received: 06.08.2025
Kabul Tarihi/Accepted: 26.01.2026



1. Giriş

Nokta bulutları havadan veya yerden kullanılan farklı tür sensörlerle elde edilebilmektedir. Havadan yapılan lazer taramalarıyla (Light Detection and Ranging, LiDAR) geniş alanların yüksek çözünürlüklü 3B nokta bulutları üretilmektedir. Deprem öncesinde ve sonrasında aynı bölgenin LiDAR taramaları gerçekleştirilmişse, bu veri kümeleri değişiklik tespiti (change detection) için kullanılabilir. Hava veya yer tabanlı fotogrametri yöntemleriyle de üç boyutlu (3B) nokta bulutları üretilmektedir. Benzer biçimde deprem öncesinde ve sonrasında elde edilen görüntülerden türetilen nokta bulutları da değişiklik tespitinde kullanılabilir. İnsansız hava araçlarında bulunan lazer tarama sensörü ve/veya kamera kullanılarak da afet sonrasında afet bölgesine ilişkin nokta bulutları üretilmektedir. Bundan başka yersel lazer tarama cihazları, stereo kameralar, RGB-D görüntüler üreten derinlik sensörleri gibi diğer kaynaklardan da nokta bulutları elde edilebilmektedir. Genelde *.las*, *.ply*, *.pcd*, *.xyz* gibi veri formatlarında kullanılan nokta bulutları, nesnelerin dış yüzeylerini tekil noktalarla temsil eden yapılandırılmamış (unstructured) bir uzamsal veri kümesidir (spatial data set) (Khoshelham & Elberink, 2012; URL-1; Wehr & Lohr, 1999).

Yapılandırılmamış nokta bulutu, her biri üç boyutlu (3B) Kartezyen dik koordinatlara (X,Y,Z) sahip bir dizi noktadan oluşmaktadır. Bu nokta bulutu yansımaya/yoğunluk değeri (intensity), renk ve doku (texture) gibi ek özellikler de bulundurulabilir. 3B nokta bulutu uzayında dağınık yapıda bulunan ve bir dizi olarak temsil edilen yapılandırılmamış nokta bulutunun, özellik uzayında yapılandırılmış şekle dönüştürülerek 3B nesneleri temsil etmesi (representation) beklenmektedir. Bununla birlikte nokta dizisinden 3B nesne temsiline geçişte nokta bulutu verisinin doğasından ve ölçme sürecinden kaynaklanan aşağıda ifade edilen bazı zorluklar bulunmaktadır:

- **Düzensizlik (irregularity):** Nokta bulutlarının görüntüyü oluşturan piksellerde olduğu gibi düzenli bir kareler ağı (grid) yapısı bulunmamaktadır. Aksine nokta bulutları düzensiz yapıda bulunmaktadır. Başka bir deyişle nokta bulutunda yer alan noktalar rastlantısal olarak dağılmaktadır. Nokta bulutlarının bu düzensiz veri yapısı (non-grid structured data); geleneksel kareler ağı veya matris tabanlı yöntemlerin nokta bulutlarında uygulanmasını zorlaştırmaktadır (M.-H. Guo vd., 2020).
- **Sırasız olma (unordered):** Nokta bulutundaki noktaların bir sırası bulunmamaktadır. Aynı geometriyi temsil eden iki ayrı nokta bulutundaki noktaların sırası farklı olabilmektedir. Nokta bulutundaki noktaların sıralı olmaması, nokta bulutunun permütasyon değişmezliği (permutation invariant) özelliği olarak da ifade edilmektedir (Liu vd., 2019; Y. Wu vd., 2024).
- **Değişken yoğunluklar (varying densities):** Nokta bulutundaki noktalar farklı yoğunluklardaki dağılımlarda (density-varying structure) bulunabilmektedir. Diğer bir ifadeyle nokta bulutunun farklı bölgelere ilişkin nokta sayıları değişkenlik göstermektedir. Bir bölgede noktalar sıkışırken yani yüksek yoğunlukta, başka bir bölgede noktalar seyrek biçimde yani düşük yoğunlukta bulunabilmektedir. Bu durum, bölütleme modellerinin hem yerel hem de global yapıyı tutarlı biçimde modellemesini zorlaştırmaktadır (Quan vd., 2024).
- **Dönme/öteleme değişmezliği (robustness to variations):** Nokta bulutunun, öteleme veya yönelim değişikliklerine karşı dayanıklı olması gereklidir.
- **Gürültülü veri kümeleri ve ölçme hataları:** LiDAR veya fotogrametri gibi ölçme teknikleri ile üretilen veri kümelerinde; gürültü, rastlantısal veya kaba hatalı gözlemler bulunabilmektedir. Bu tür bozulmalar, nokta bulutlarındaki bölütleme görevinin sonuçlarının doğruluğunu olumsuz yönde etkilemektedir.
- **Çoklu ölçek ve çok boyutluluk:** Nokta bulutları, farklı ölçeklerde ayrıntılar barındırabilmektedir. Küçük ölçekli

detayların yanı sıra, büyük ölçekli yapısal özelliklerin de korunması ve doğru şekilde ayrıştırılması gerekmektedir. Bu durum, yapay öğrenme modellerinin çok ölçekli özellik çıkarım yeteneğine sahip olması gerektirmektedir (Wijaya vd., 2024).

- **Yerel ve global özelliklerin bütünleştirilmesi:** Nokta bulutunun hem yerel ayrıntılarını hem de global yapısını etkin biçimde belirleyebilmek, bölütleme modelleri için kritik bir problemdir. Yerel ilişkiler ve global bağlam arasındaki dengeyi kurmak, bölütleme modelinin genel başarımını belirleyen diğer bir önemli etkidir (Z. Lin vd., 2024).
- **Hesaplama karmaşıklığı:** Deprem sonrası durumu temsil eden büyük ve karmaşık nokta bulutu veri kümeleri üzerinde bölütleme yaparken, yüksek hesaplama maliyetleri ve bellek gereksinimleri ortaya çıkmaktadır. Gerçek zamanlı veya yarı gerçek zamanlı uygulamalarda bu durum başarımları kısıtlamalarına neden olabilmektedir.

Bu zorluklar, nokta bulutlarının bölütleme görevinde kullanılacak olan yapay öğrenme mimarilerinin tasarımında dikkate alınması gereken temel unsurlardır. Nokta bulutu bölütleme görevinde büyük ilerlemeler kaydedilmiş olsa da deprem sonrası gibi karmaşık gerçek dünya ortamlarında (complex real-world environments) güvenilir başarımlar sağlamak için hâlâ çözülmesi gereken birçok güçlük bulunmaktadır. Başarılı bir bölütleme görevi için verinin düzensiz yapısı, sıralı olmaması, değişken yoğunlukları, gürültüsü ve çok ölçekli özellikleri gibi unsurları uygulama gereksinimlerine uygun olacak biçimde dikkate alarak, noktalardan anlamlı özellikleri çıkarabilecek özel mimari tasarımların yapılması gerekmektedir.

2. Nokta Bulutlarında Özellik Çıkarım Mimarileri

Yapılandırılmamış nokta bulutu veri kümelerinden anlamlı temsiller elde edebilmek için özellik çıkarım mimarileri tasarlanmakta ve geliştirilmektedir. Anlamlı bir temsil, yapay öğrenme modellerinin nokta bulutlarını daha kolay anlayabilmesini ve üzerinde daha etkili bir şekilde çalışabilmesini sağlamaktadır. Nokta bulutlarından özellik çıkarım yaklaşımlarıyla oluşturulan temsiller daha sonra nesne tespiti, sınıflandırma, bölütleme gibi bilgisayarlı görü görevlerinde kullanılmaktadır. Nokta bulutlarında özellik çıkarım mimarileri hem klasik yaklaşımlarla hem de derin öğrenme temelli yaklaşımlarla oluşturulabilmektedir.

Klasik yaklaşımlarda aşağıda ifade edilen geometrik ve yerel tanımlayıcılar ile çoklu özellikli yaklaşımlar kullanılmaktadır:

- **Geometrik tanımlayıcılar:**
 - Normal Vektörler (surface normals) (Hoppe vd., 1992)
 - Eğrilik (curvature) ve Düzlemsellik (planarity) (URL-2)
- **Yerel tanımlayıcılar:**
 - Her noktanın etrafındaki yerel geometri; Spin Images (Johnson & Hebert, 1999), 3D Shape Contexts (Frome vd., 2004), Fast Point Feature Histograms (FPFH) (Rusu vd., 2009) ve Signature of Histograms of Orientations (SHOT) (Tombari vd., 2010) gibi histogramlar veya benzeri tanımlayıcılarla özetlenir.
- **Çoklu ölçekli (multi-scale) yaklaşımlar:**
 - Küçük ve geniş komşuluk pencereleri gibi farklı ölçeklerde yapılan analizlerle hem yerel hem de global yapısal detaylar elde edilir. Bu, özellikle karmaşık yapıların analizinde önemlidir ve derin öğrenme mimarilerinde 'çok ölçekli özellik çıkarımı (multi-scale feature extraction)' katmanları ile benzer mantıkta uygulanır (Qi vd., 2017b).

Derin öğrenme yaklaşımıyla özellik çıkarımında; yapay öğrenme modeli nokta bulutlarındaki karmaşık geometrik ilişkileri kendisi doğrudan öğrenerek hem yerel hem de global yapısal detayları verimli biçimde belirleyebilmektedir. Böylece klasik yaklaşımlara göre daha zengin anlamlı temsiller oluşturabilmektedir. Nokta bulutlarında derin öğrenme yaklaşımıyla özellik çıkarımında kullanılan temel yöntemler genel olarak aşağıda ifade edilmiştir:

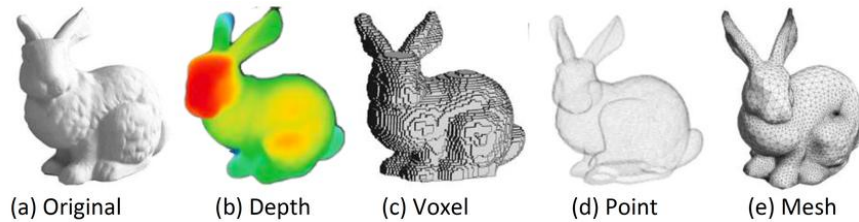
- Nokta tabanlı (pointwise) yöntemler (Qi vd., 2017a; Qi vd., 2017b),
- Evrişim işlemleri (convolution operations) tabanlı yöntemler (J. Li vd., 2018; Thomas vd., 2019),
- Çizge tabanlı (graph-based) yöntemler (Wang vd., 2018),
- Dönüştürücü (transformer) tabanlı yöntemler (H. Zhao vd., 2021).

Seçilen mimari ve/veya model; verinin büyüklüğüne, ayrıntı gereksinimine ve hesaplama kaynaklarına bağlıdır. Son yıllarda çizge tabanlı ve dönüştürücü temelli modeller karmaşık ilişkileri modellemede diğer yöntemlere üstünlük sağlamaktadır (Chen vd., 2024; Lu vd., 2022).

2.1 3B Özellik Çıkarım Ağları

Görüntüler üzerinde kullanılan derin öğrenme modellerinin doğrudan nokta bulutlarına uygulanması bazı nedenlerden dolayı mümkün değildir. İlk olarak raster veri yapısında olan görüntülerin piksellerden oluşan düzenli bir temsili bulunmaktadır. Vektör veri yapısında bulunan nokta bulutları ise taraması yapılan ortamı seyrek, aralıklı şekilde ve değişken nokta yoğunluğuyla temsil etmektedir. Diğer bir farklılık ise nesne tanımlama, bölütleme, çıkarma gibi bilgisayarlı görü kavramları görüntüler üzerinden iki boyutlu gerçekleştirilirken nokta bulutlarında bu görevler üç boyutlu olarak gerçekleştirilmektedir (URL-3).

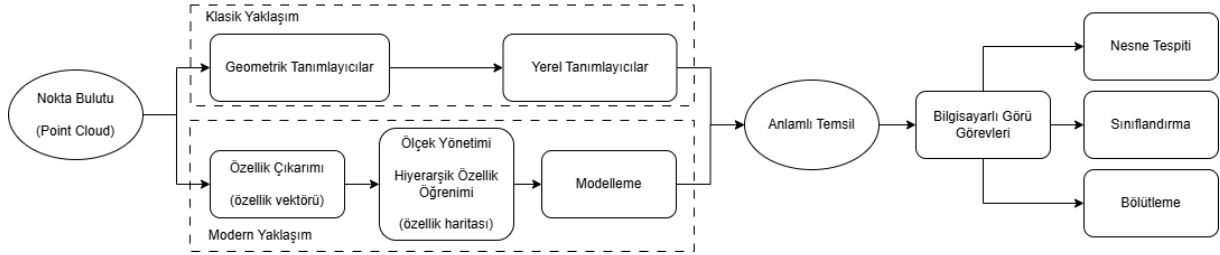
Bu nedenle nokta bulutlarında kullanılan evrişim işlemleri (convolution operations in point clouds), görüntüler üzerinde kullanılan geleneksel ‘Evrişimli Sinir Ağları (Convolutional Neural Network, CNN)’ yaklaşımından farklılık göstermektedir. Görüntülerde pikseller düzenli bir kareler ağı (grid) yapısında bulunmakta ve evrişim çekirdekleri (convolutional kernel) bu düzenli ve sıralı yapı üzerinde kaydırılarak (sliding window) özellikler çıkarılmaktadır. Nokta bulutlarının düzensiz, sırasız ve seyrek yapısı geleneksel filtrelerin uygulanmasını zorlaştırmaktadır. Nokta bulutlarında özellik çıkarmak için kullanılan evrişim işlemleri, nokta bulutlarının bu düzensiz veri yapısına uyum sağlamak için özel mimariler yani ‘3B Özellik Çıkarım Ağları (3D Backbone Network)’ gerektirmektedir.



Şekil 1: Stanford tavşan modelinin 3B veri formları (Chen vd., 2023; Hoang vd., 2019)

3B Özellik Çıkarım Ağları; Şekil 1’de gösterilen nokta bulutu, 3B piksel (voxel) veya mesh (üçgen/poligon ağı) gibi 3B veri yapılarından temel özellikleri/detayları (features) çıkarmak ve 3B veri kümesinden anlamlı temsil öğrenmek için tasarlanmış derin öğrenme mimarileridir. Bu ağlar, nokta bulutundaki önemli bilgiyi ve örüntüleri yakalamalı, gürültüyü azaltmalı ve sınıflandırma, bölütleme gibi sonraki bilgisayarlı görü görevleri için daha faydalı bir yapıyı/temsili ortaya çıkarmalıdır. Şekil 2’de görüldüğü üzere 3B Özellik Çıkarım Ağı temel olarak üç bileşenden oluşmaktadır. Özellik çıkarım bileşeninde yüzey

eğriliği, kenar gibi nokta bulutundaki geometrik yapılar veya bina gibi bağlamsal bilgiler çıkarılmaktadır. Ölçek yönetimi bileşeninde çok ölçekli özellikler çıkarılmaktadır. Bu bileşende hiyerarşik özellik öğrenilmekte ve küçük ölçekteki yerel detaylar ile büyük ölçekteki global detaylar arasında denge kurulmaktadır. Modelleme bileşeninde ise nokta bulutlarının sırasız ve düzensiz yapısından anlamlı temsil oluşturulmaktadır.



Şekil 2: 3B özellik çıkarma ağ yapısı

3B Özellik Çıkarma Ağları, nokta bulutunun dağılımını ve yapısal özelliklerini otomatik olarak öğrenmektedir. Bu şekilde yapay öğrenme modeli; nokta bulutunun düzensizliği, sıralı olmaması, seyrekliği ve gürültüsü gibi nokta bulutunun kendine özgü zorluklarına karşı daha dayanıklı hale gelmektedir. 3B Özellik Çıkarma Ağları, anlamlı temsil oluşturmak için genellikle ‘Çok Katmanlı Algılayıcı (Multi-Layer Perceptron, MLP)’ ve/veya nokta bulutlarına özgü evrişim işlemlerini hiyerarşik bir şekilde kullanmaktadır. MLP, noktaların koordinatlarını ve diğer özelliklerini daha yüksek boyutlu bir özellik uzayına dönüştürmektedir. Bu işlem, nokta bulutundaki tüm noktalar için paralel olarak uygulanmaktadır (URL-4).

‘Pointwise MLP’, her noktayı bağımsız olarak işlediği için hızlı ve verimlidir, ancak yerel bağlamı belirlemede eksik kalmaktadır. Diğer bir ifadeyle, komşu noktalar gibi noktalar arasındaki yerel ilişkileri dikkate almadığından uzamsal ilişkileri doğrudan modelleyememektedir (H. Zhao vd., 2021).

Evrişim işlemlerinde girdideki belirli özellikleri tespit eden veya baskın özellikleri öne çıkaran evrişim çekirdekleri (convolutional kernel) kullanılmaktadır. Evrişim işleminin sonucunda, girdinin farklı bölgelerinde tespit edilen özelliklerin ne kadar güçlü bir şekilde belirlendiğini gösteren bir ‘Özellik Haritası (Feature Map)’ ortaya çıkmaktadır (Krizhevsky vd., 2017).

Nokta bulutundaki tüm noktaların yerel ve global özelliklerini birleştirmek için yani global bir temsil oluşturmak için maksimum havuzlama (maximum pooling), ortalama havuzlama (average pooling) veya toplam havuzlama (sum pooling) gibi simetrik fonksiyonlar kullanılmaktadır. 3B özellik çıkarma ağlarında bu mekanizma ‘Global Özellik Toplama (Global Feature Aggregation)’ olarak isimlendirilmektedir. Global özellik toplama sonucunda ortaya çıkan global özellik vektörü (global feature vector) ile nokta bulutunun daha zengin bir temsili elde edilmektedir. Havuzlama aşamasında yalnız en aktif nöronlar bir sonraki katmana taşınmak üzere seçildiği için özelliklerin bir kısmı göz ardı edilmekte yani kullanılmamaktadır. Bu durum bazı uzamsal bilgilerin katmanlar arasında kaybolmasına neden olmaktadır (Goodfellow vd., 2016; LeCun vd., 1998).

Buraya kadar ifade edilen kısım, nokta bulutlarının düzensiz ve sırasız yapısına uyum sağlamak için oldukça basit ve verimli bir yaklaşım olarak kabul edilebilir. Bununla birlikte, karmaşık yapılardan özellik çıkarmak veya nesne sınırlarını iyi ayırt edebilmek için yeterli olmayacaktır. Bu nedenle ek mekanizmalarla desteklenmesi gerekmektedir.

Derin öğrenme mimarilerinin, özellikle ‘Derin Sinir Ağları (Deep Neural Network, DNN)’ gibi çok katmanlı yapay sinir ağlarının, girdi veri kümesinde basitten karmaşığa doğru artan seviyelerde özellikleri çıkararak ilerlemesi bir tür hiyerarşik

özellik öğrenmedir. 3B özellik çıkarım ağının hiyerarşik yapısı; farklı ölçeklerdeki ve soyutlama seviyelerindeki özelliklerin öğrenilmesini sağlamaktadır. Örneğin, ilk katmanlar daha yerel ve ince detayları yakalarken, daha derin katmanlar daha global ve semantik bilgiyi temsil etmektedir. 3B özellik çıkarım ağı; yerel temsilleri hiyerarşik olarak birleştirilerek hem yerel hem de global özellikler elde edilmesini ve verinin daha zengin bir temsili oluşturulmasını sağlamaktadır. Böylece detaylı bölütleme, karmaşık nesne tanıma gibi görevler için daha etkili bir temsil elde edilmiş olacaktır.

Çalışma kapsamında odaklanılmış olan nokta bulutu bölütleme görevinde temel amaç; nokta bulutundaki noktaların çevrelerindeki yüzeyleri, nesneleri veya yapıları temsil eden belirgin anlamsal parçalara kümelenebilir. Başka bir deyişle, nokta etiketlerini (point labels) bina, yol gibi aynı gerçek dünya ögesine ait nokta gruplarına sınıflandırabilmektir (He vd., 2021; M.-H. Guo vd., 2020). Bunun için 3B özellik çıkarım ağlarının (backbone) çıkardığı zengin özellik haritalarının üzerine, nokta bazında sınıflandırma yapacak bir 'bölütleme başlığı' (segmentation head) eklenmektedir. Bölütleme başlığı her bir noktanın hangi sınıfa ait olduğunu belirlemekte ve her bir nokta için bina, yol, arka plan gibi bir sınıf etiketi tahmin etmektedir.

3. Nokta Bulutu Bölütlemesi

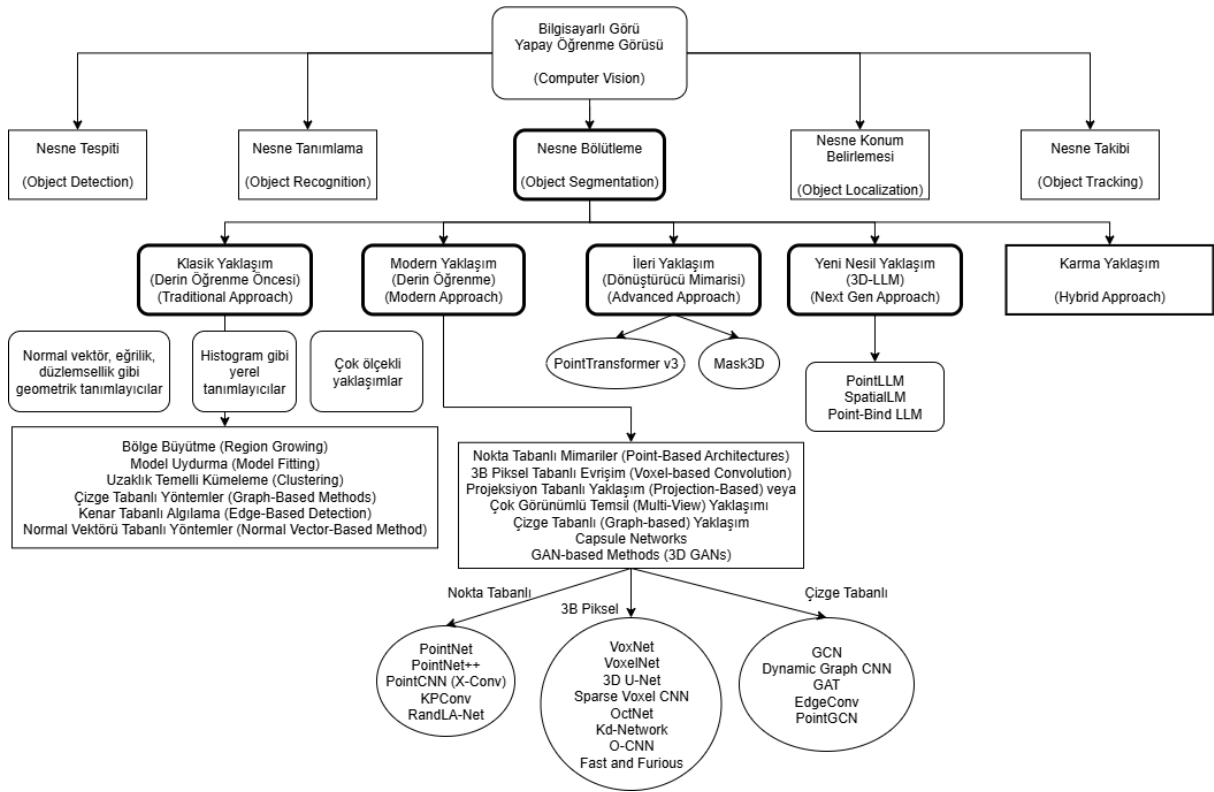
Nokta Bulutu Bölütlemesi (Point Cloud Segmentation) için kullanılabilecek olan yöntemler bu çalışma kapsamında aşağıdaki gibi dört ana gruba ayrılmış ve Şekil 3'te gösterilmiştir:

- **Klasik yaklaşım (traditional approach):** Derin öğrenme öncesi kullanılan geleneksel nokta bulutu bölütleme yöntemleri
- **Modern yaklaşım:** Özellikle CNN olmaya üzere derin öğrenme tabanlı yöntemler (deep learning-based approaches)
- **İleri derin öğrenme (advanced deep learning) yaklaşımı:** Dönüştürücü temelli (transformer-based) yenilikçi yöntemler
- **Yeni nesil yaklaşımlar (next-gen approaches):** Büyük Dil Modellerine (LLM) uzamsal anlayış kazandıran ve bu dil modellerini 3B uzamsal veri kümeleri ile bütünleştiren çok modlu (multi-modal) Üç Boyutlu Büyük Dil Modeli (3D-Large Language Model, 3D-LLM) temelli yöntemler

Bir de bu yaklaşımları birlikte kullanabilen karma (hybrid) yaklaşımlar bulunmaktadır. Karma yaklaşım, klasik ve derin öğrenme yaklaşımlarını ya da diğer yenilikçi yaklaşımları birleştirerek her iki yöntemin üstünlüklerinden yararlanmayı hedeflemektedir. Örneğin, klasik yöntemlerle ön bölütleme yapılabilir ve ardından derin öğrenme modelleri ile bu bölütlemeler iyileştirilebilir.

3.1 Klasik Nokta Bulutu Bölütlemesi

Derin öğrenme öncesi kullanılan geleneksel nokta bulutu bölütlemesi yöntemleri genellikle geometrik özelliklere, istatistiksel analizlere ve kümeleme algoritmalarına dayanmaktadır. Klasik nokta bulutu bölütlemesi diğer yöntemlere göre daha az veri kümesi gerektirirken, karmaşık sahneleri bölütleme görevinde sınırlı başarımlar gösterebilmektedir. En yaygın kullanılan klasik bölütleme yaklaşımları Şekil 3'te belirtilmiştir.



Şekil 3: Nokta bulutu bölütleme yaklaşımları

3.2 Modern Nokta Bulutu Bölütlemesi

Derin öğrenme tabanlı yöntemler, büyük veri kümelerinden karmaşık özellikleri öğrenebilmektedir. Bu yöntemler nokta bulutunu veya 3B piksel (voxel) ya da mesh (üçgen/poligon ağı) gibi onun farklı gösterimlerini girdi olarak almaktadır. Ardından geleneksel CNN mimarilerinin gücünden yararlanmaktadır. Başka bir deyişle nokta bulutlarının düzensiz yapısı üzerinde çalışabilecek derin öğrenme mimarilerinin tasarlanması ve modellerin geliştirilmesi üzerine yoğunlaşmıştır. Modern nokta bölütlemesi yöntemleri, klasik yöntemlere göre daha yüksek başarımlar sağlamaktadır. Bununla birlikte büyük miktarda etiketlenmiş veri kümesi ve güçlü hesaplama kaynakları gerektirmektedir.

Çalışma kapsamında modern bölütleme her ne kadar CNN-temelli olarak ifade edilmiş olsa da Kapsül Ağları (Capsule Networks), ‘Çekişmeli Üretici Ağları (Generative Adversarial Network, GAN)’ ve Otomatik Kodlayıcıları (Auto Encoders, AE) kullanan mimariler de bulunmaktadır. PointCapsNet (Sabour vd., 2017; Y. Zhao vd., 2019), 3D GAN olarak kabul edilebilecek LatentGAN (Learning Representations and Generative Models for 3D Point Clouds) (Achlioptas vd., 2017) CNN-temelli olmayan mimarilere örnek olarak gösterilebilir.

3.2.1 Nokta Tabanlı Mimariler (Point-Based Architectures)

Nokta tabanlı mimariler nokta bulutunu doğrudan girdi olarak almaktadır. Çok katmanlı algılayıcılar (MLP) ve evrişim işlemleri ile her noktayı ayrı ayrı işlemekte ve bölütleme maskesini tahmin etmektedir.

Aşağıda detaylı ifade edilen mimariler dışında A-CNN, RS-CNN (Relation-Shape CNN), PointConv, GeoConv, SFCNN (Spherical Fractal CNN), RSNet gibi birçok farklı CNN temelli mimari bulunmaktadır.

PointNet (2017): PointNet, nokta bulutundaki noktaların permütasyon değişmezliğine dikkat eden ve MLP ile T-Net kullanarak sıralanmamış nokta bulutlarını doğrudan işleyebilen ilk derin öğrenme mimarilerinden biridir. Her noktayı

bağımsız olarak işlemekte, çok katmanlı algılayıcılar ile özelliklerini çıkarmakta ve havuzlama katmanı (max pooling) ile bir global özellik vektörü oluşturmaktadır. Global özellik vektörü ile her noktanın bireysel özellikleri birleştirilerek ve aynı zamanda her nokta için bireysel bir etiket tahmin edilerek bölütleme görevi gerçekleştirilmektedir. Nokta sıralamasına duyarlı olmayan yapısı sayesinde düzensiz nokta bulutlarıyla doğrudan çalışabilmektedir (Qi vd., 2017a).

Basit ve verimli bir yöntem olarak kabul edilmektedir. Ancak özellikle karmaşık sahnelerde yerel bağlam bilgisini yani noktalar arası ilişkileri etkili bir şekilde yakalamakta zorlanabilmektedir. Bu yaklaşım her ne kadar ayrıntıları koruyabilse de yerel bağlam modellemesinde eksiklikler barındırmaktadır. PointNet++ bu sorunu kısmen çözmektedir (Qi vd., 2017b).

PointNet mimarisinden bağımsız olarak geliştirilen öncü derin öğrenme yaklaşımlarından bir diğeri ‘Deep Sets’ mimarisidir. Her ne kadar benzer yaklaşım gösterebilir de iki mimari arasındaki temel fark MLP ağındaki ağırlık paylaşımıdır (Zaheer vd., 2017).

PointNet++ (2017): PointNet++, PointNet yönteminin geliştirilmiş bir sürümüdür. Bu yaklaşım kümeleme ve alt örnekleme gibi hiyerarşik özellik öğrenme ilkelerini uygulayarak farklı ölçeklerdeki yerel yapıları/özellikleri çıkarabilmektedir. İlk aşamada yerel bölgelerden yani belirli bir yarıçap içindeki komşu noktalardan özellik çıkarımı yapmaktadır. Ardından çıkardığı bu yerel özellikler birleştirilerek global özellik temsili oluşturmaktadır. Böylece çok ölçekli (multi-scale) bilgilerin bütünleşmesi sağlanarak yerel ve global özellikler daha iyi belirlenmektedir. Başka bir deyişle hem yerel özellikleri özetleyebilmekte hem de kapsamlı bir özellik temsili oluşturabilmektedir (Qi vd., 2017b).

PointNet yöntemine göre yerel yapıları daha iyi belirlemekte ve daha iyi bölütleme başarımı ortaya çıkarmaktadır. Bununla birlikte hesaplama maliyeti ve model karmaşıklığı PointNet yönteminden daha yüksektir.

PointNet++ öncü mimarisine dayalı olarak geliştirilmiş ‘Structural Relation Network (SRN)’ (Duan vd., 2019), ‘Adaptive Feature Adjustment (AFA)’ destekli PointWeb (H. Zhao vd., 2019), ‘Self-Organizing Map (SOM)’ temelli ‘Self-Organizing Network (SONet)’ (J. Li vd., 2018) gibi başka modeller de bulunmaktadır.

PointCNN (X-Conv) (2018): PointCNN yönteminde, önce nokta bulutunun düzensiz yapısı X-Conv evrişim operatörünü kullanılarak düzenli yapıya dönüştürülmektedir. Bu operatör, noktaların yerel komşuluklarını ve özelliklerini bir matris dönüşümü ile öğrenmekte ve evrişimsel bir şekilde işlemektedir. Ardından geleneksel CNN mimarisi ile evrişim işlemlerini uygulayarak yerel yapısal bilgiyi (yerel ilişkileri) çıkarmaktadır. Yerel geometrik bilgiyi etkili bir şekilde çıkardığı için de daha iyi bölütleme sonuçları elde edebilmektedir (J. Li vd., 2018).

PointNet mimarisinde özellik çıkarım aşamasında maksimum havuzlama gibi simetrik fonksiyon kullanılırken, PointCNN mimarisinde uygulamalarda daha etkin olduğu görülen ve kanonik sıralamaya (canonical order) dayalı çalışan X-Conv evrişim operatörünü kullanılmaktadır (J. Li vd., 2018). Bununla birlikte X-Conv evrişim operatörünün öğrenilmesi ek hesaplama maliyeti getirmektedir. Bu nedenle hesaplama maliyeti PointNet ve PointNet++ yaklaşımlarına göre daha yüksek olabilmektedir. Ayrıca dönüşüm işlemi bilgi kaybına yol açabilmektedir.

Kernel Point Convolution (KPConv) (2019): KPConv yöntemi, nokta bulutları üzerinde evrişim işlemini çekirdek noktalar (kernel points) kullanarak gerçekleştirmektedir. Her nokta etrafında belirli bir yarıçap içinde bulunan komşu noktalar için çekirdek noktalar yerleştirilmektedir. Bu çekirdek noktalarla evrişim benzeri işlemler gerçekleştirilerek geleneksel evrişim ağırlıklarına benzer şekilde yerel özellikler öğrenilmektedir. Çekirdek noktalarının ağırlıkları, komşu noktaların konumlarına göre belirlenmektedir (Thomas vd., 2019). Başka bir deyişle, KPConv yöntemi, geleneksel evrişim işleminin uzamsal yakınlık ve ağırlıklandırma prensiplerini, nokta bulutlarının düzensiz geometrik yapısına uyarlayarak esnek bir yapı

sunmaktadır. Yerel geometrik detayları çok iyi çıkarabildiği için yüksek başarımlı bölütleme gerçekleştirilebilmektedir. Çekirdek noktalarının konfigürasyonu ve optimizasyonu nokta bulutuna bağlı olarak ek ayar gerektirmesi, hesaplama maliyetini yükseltebilmektedir.

RandLA-Net (2020): RandLA-Net yöntemi büyük ölçekli nokta bulutlarını işlemek için rastlantısal örnekleme (random sampling) ve yerel özellik toplama düzeneklerini kullanmaktadır (Hu vd., 2020a). Hafif ve hızlı olması nedeniyle büyük veri kümeleri için ideal olarak kabul edilebilir. Bununla birlikte rastlantısal örnekleme nedeniyle ayrıntı kaybı olabilmektedir.

3.3 İleri Derin Öğrenme Temelli Nokta Bulutu Bölütlemesi

İleri derin öğrenme temelli nokta bulutu bölütlemesi yaklaşımında, dönüştürücü temelli (transformer-based) derin öğrenme mimarilerinin nokta bulutlarındaki düzensiz ve sırasız yapıları ele alma yeteneğinden faydalanılmaktadır. Bu mimariler, noktalar arası ilişkileri dikkat (self-attention) veya dönüştürücü (transformer) mekanizmalarıyla modellemekte ve önemli noktaların ağırlıklarını artırmaktadır. Dönüştürücü temelli yöntemler, CNN temelli yöntemlerden farklı olarak uzun mesafeli bağımlılıkları iyi biçimde belirlemekte, nokta bulutundaki yerel ve global ilişkileri etkili şekilde öğrenmekte ve böylece daha yüksek başarımlı sağlamaktadır. Bununla birlikte, büyük veri kümelerine ve yüksek hesaplama gücüne gereksinim duymaktadır.

Dönüştürücü mimarisi genel olarak kodlayıcı-kod çözücü (encoder-decoder) yapısını kullanmaktadır. Kodlayıcı girdi verisine çeşitli dönüşümler uygulayarak girdi verisini anlamlı bir temsile dönüştürmektedir. Kod çözücü, kodlayıcı tarafından üretilen temsili kullanarak hedef çıktıyı üretmektedir. Dönüştürücü mimarisi dikkat düzeneklerini katmanlar halinde kodlayıcı-kod çözücü ve 'konum kodlama (positional encoding)' yapıları ile birlikte kullanarak bölütleme gibi karmaşık görevleri yerine getirebilmektedir. Kodlayıcı nokta bulutu gibi bir girdi veri kümesini öz dikkat ve ileri besleme (feed forward) katmanlarıyla kodlarken, kod çözücü; kodlayıcının çıktısını ve belirli bilgileri kullanarak yapılandırılmış nokta bulutunu üretmektedir. Konum kodlama, düzensiz yapıdaki nokta bulutundaki noktaların konum bilgisini kodlamaktadır. Alt örnekleme ve üst örnekleme süreçleri, çok ölçekli analizi olanaklı hale getirmekte ve bölütleme gibi bilgisayarlı görü görevlerinin başarımlarını arttırmaktadır (Lu vd., 2022).

Dikkat düzeneği (attention mechanism), modelin girdinin belirli bölümlerine odaklanmasını sağlayan bir bileşen olup ilişkileri yakalamak için kullanılmaktadır. Bu sorgu (query), anahtar (key) ve değer (value) vektörleriyle hesaplanmakta ve ardından bir dikkat ağırlık matrisi oluşturulmaktadır. Bu ağırlıklar, her noktanın diğer noktalardan ne kadar bilgi alacağını belirlemektedir (Vaswani vd., 2017).

'Öz Dikkat (Self-Attention)' ve 'Çok Başlıklı Öz Dikkat (Multi-Head Self-Attention)', dönüştürücü mimarisinin temelini oluşturan ve bir dizideki farklı konumlar arasındaki ilişkileri modellemeye yarayan düzeneklerdir. Aralarındaki temel fark; çok başlı öz dikkatin, dikkat işlemini paralel olarak birden fazla başlık (head) üzerinden gerçekleştirmesidir. Böylece tek bir öz dikkat düzeneğinin yani tek başlıklı öz dikkat düzeneğinin yakalamada güçlük çekebileceği farklı türdeki ilişkiler ve bilgiler çok başlı öz dikkat düzeneği ile yakalanabilmektedir. Çünkü her bir başlık, veri kümesindeki farklı bir bakış açısına odaklanarak daha kapsamlı bir temsil elde edilmesini sağlamaktadır. Diğer bir ifadeyle, öz dikkat temel ilişkiyi öğrenmekte ancak sınırlı bir bakış açısı sunmaktadır. Çok başlı öz dikkat ise karmaşık yapıları çoklu bakış açısıyla daha iyi modelleyebilmektedir. Bu da yapay öğrenme modelinin daha karmaşık örüntüleri ve bağımlılıkları anlamasına yardımcı olmaktadır. Çok başlı dikkat düzenekleri, dönüştürücü mimarisinin başarısının temelini oluşturmaktadır (H. Zhao vd., 2021; Vaswani vd., 2017).

Buradaki başlık (head) kavramı, dikkat düzeneğinin tek bir paralel işlem birimi olarak anlaşılmalıdır. Her bir başlık, bağımsız

olarak ayrı bir dikkat düzeneği gibi davranmaktadır. Her bir başlık; dikkat düzeneğini paralel biçimde farklı alt uzaylarda yani farklı bakış açılarında çalışmasını ve böylece girdi nokta bulutundaki farklı özelliklerin/detayların çıkarılmasını sağlamaktadır. Her bir başlık; veri kümesinin farklı bir boyutunu veya veri kümesindeki farklı bir ilişki türünü öğrenmektedir. Sonuçta elde edilen bilgiler birleştirilerek daha zengin ve kapsamlı bir temsil oluşturulmaktadır.

Sözü edilen her iki dikkat düzeneği de nokta bulutu gibi düzensiz veri yapılarında uzun mesafeli bağımlılıkları yakalamak için kullanılmaktadır. Deprem hasar tespitinde, tek başlıklı öz dikkat düzeneği kullanıldığında binanın genel çökme şekli algılanabilir. Çok başlıklı öz dikkat düzeneği kullanıldığında ise birinci başlık yerel çatlakları tespit edebilir, ikinci başlık binanın genel stabilitesini analiz edebilir ve üçüncü başlık enkaz parçalarıyla bina arasındaki ilişkiyi öğrenebilir. Yani çok başlıklı öz dikkat düzeneği ile hasarlı bir binanın hem detayları hem de bütünü daha iyi modellenebilir.

3.3.1 PointTransformer (PTv1) (2020)

PointTransformer mimarisi, 3B nokta bulutlarının düzensiz yapısını işleyebilmek için özel olarak tasarlanmış ilk dönüştürücü temelli mimaridir. Nokta bulutlarını hiyerarşik bir şekilde işlemek için katmanlar halinde yerel dikkat düzeneklerini uygulamakta ve nokta sayısını azaltma tekniklerini (downsampling) kullanmaktadır (H. Zhao vd., 2021). Bu mimari, çıktı olarak her nokta için bağlam bilgisi (context-aware) özellik vektörleri üretmektedir.

PointTransformer mimarisinin temel bileşenleri aşağıdaki gibidir (H. Zhao vd., 2021):

- **Özellik soyutlama katmanı (feature abstraction layer):** Girdi nokta bulutundaki her nokta için ilk özellikler çok katmanlı algılayıcı (MLP) kullanılarak çıkarılır. Nokta bulutundaki her bir noktanın 3B koordinatları (X,Y,Z) ve varsa yüzey normaleri, yansıma değeri, renk gibi ek özellikleri eklenerek özellik vektörleri oluşturulur (input embedding). Bu özellikler, genellikle çok katmanlı algılayıcılar gibi lineer dönüşümlerle daha yüksek boyutlu bir özellik uzayına yansıtılır. Ardından bu özellikler öz dikkat düzenekleriyle birleştirilir.
- **Öz dikkat katmanı (self-attention layer):** Komşu noktalarla olan ilişkileri öğrenmek için nokta bulutundaki her bir nokta, öz dikkat düzeneği aracılığıyla diğer noktalarla etkileşime girer. Öz dikkat katmanında her noktanın, diğer tüm noktalarla olan ilişkileri öğrenilir. Bu ilişkiler ağırlıklandırılarak uzun mesafeli bağımlılıklar belirlenir. Her nokta için sorgu, anahtar, değer vektörleri ve ardından bir dikkat ağırlık matrisi oluşturulur. Bu ağırlıklar, her noktanın diğer noktalardan ne kadar bilgi alacağını belirler. Her nokta için, diğer tüm noktalardan ağırlıklı bir özellik toplamı hesaplanır. Bu, her noktanın bağlamını anlamasına ve daha iyi özellikler öğrenmesine yardımcı olur.
- **Konum kodlama (positional encoding):** Noktaların uzamsal konum bilgisini dikkat mekanizmasına dâhil etmek ve nokta bulutunun yerel bölgelerindeki uzamsal ilişkilerini modellemek için kullanılır.
- **Yerel özellik toplama (feature aggregation):** Yerel komşuluk bölgeleri dikkate alınarak hem yerel hem de global özelliklerin birleşik temsili oluşturulur.
- **Hiyerarşik özellik öğrenme (hierarchical feature learning):** Yerel bölgelerdeki ilişkiler ve global bağlam hiyerarşik olarak öğrenilmektedir. Nokta bulutunun farklı ölçeklerdeki özellikleri aşamalı olarak öğrenilir. Nokta bulutundaki yerel ve global bilgilerin çok katmanlı olarak öğrenilmesini sağlayan hiyerarşik yaklaşım; ‘katmanlı/aşamalı özellik öğrenimi’ ve ‘kodlayıcı-kod çözücü’ olmak üzere iki ana bileşenden oluşmaktadır.
 - Katmanlı özellik öğrenimi bileşenin ilk katmanlarında nokta bulutu kNN, Farthest Point Sampling (FPS) ve yarıçap gibi komşuluk ilişkileri kullanılarak küçük gruplar halinde işlenir. Bu aşamada her noktanın çevresindeki komşular üzerinden yerel özellikler çıkarılır. Katmanlı özellik öğrenimi bileşenin ilerleyen katmanlarında yerel özellikler daha geniş bir alanda global olarak birleştirilir. Yani, daha önce çıkarılan

- yerel özellikler, daha büyük gruplar veya alt kümeler üzerinden özetlenerek, tüm nesnenin veya sahnenin genel yapısal özellikleri elde edilir.
- Kodlayıcı-kod çözücü bileşeninin kodlayıcı kısmında nokta bulutu, hiyerarşik yapıda işlenmek üzere çeşitli ölçeklerde alt bölümlere ayrılır. Her aşamada, veri nokta sayısı azaltılır (downsampling) ve yerel dikkat mekanizmaları ile özellik çıkarımı yapılır. Kod çözücü kısmında ise kodlayıcıdan gelen hiyerarşik özellikler orijinal çözünürlüğe (upsampling) getirilir. Böylece hem yerel detaylar korunur hem de global bağlam yeniden yapılandırılır.
- **Normalleştirme ve artık bağlantılar (normalization and residual connections):** Noktaların sıralı olmayan yapısını dengelemek ve ağırların eğitimi kolaylaştırmak için çeşitli normalleştirme teknikleri kullanılır.

PointTransformer v1 mimarisi karmaşık yapılarda global bağlamı etkili biçimde modellemesine karşın kenar, yüzey gibi yerel geometrik detayları modellemekte zayıf kalabilmektedir. Ayrıca tüm noktalar arası ilişkileri hesaplandığından dolayı özellikle büyük nokta bulutu veri kümeleri için hesaplama maliyeti yüksektir.

3.3.2 PointTransformer v2 (PTv2) (2021)

PointTransformer v2 mimarisinde hem model başarımını hem de hesaplama verimliliğini arttırmak için PTv1 modelindeki bazı mimari eksiklikler giderilmiştir. Her şeyden önce yerel özellikleri daha iyi belirlemek ve yerel-global bağlam dengesini daha iyi kurmak için dinamik bir yerel dikkat düzeneği ve gruplandırılmış vektör dikkati kullanılmıştır (X. Wu vd., 2022).

Point Transformer mimarisindeki tam dikkat düzeneği yerine PTv2 mimarisinde dinamik yapıdaki yerel dikkat (local attention) düzeneği kullanılmaktadır. Yerel dikkat düzeneği, her bir noktanın yalnızca belirli bir komşuluk kümesi üzerinde dikkat hesaplaması yapmasını sağlamaktadır. Söz konusu yerel dikkat düzeneği, orijinal mimarideki gibi 'gruplandırılmış dikkat (grouped attention)' gibi ön tanımlı gruplar veya statik bölümler kullanmamaktadır. Komşuluklar her nokta için ayrı ayrı ve dinamik olarak hesaplanmaktadır. Her nokta, yalnız özellik benzerliğine göre belirli komşularıyla etkileşime girerek yerel özellikleri öğrenmektedir. Her nokta komşularını k en yakın komşu gibi konumlarına göre belirlemekte ve bu bölge içinde 'ofset dikkat (offset attention)' düzeneğini uygulamaktadır. 'Ofset dikkat (offset/relative attention)' düzeneği, her komşuluk kümesindeki noktalar arasındaki ilişkilerin merkez/referans noktasına olan uzaklıklarını/ofsetlerini dikkate alarak, noktaların konumlarının ve özelliklerinin birlikte öğrenilmesidir. Ofset dikkat düzeneği, yerel geometrik bilginin daha iyi öğrenilmesine yardımcı olmaktadır. Ayrıca komşu noktaların ağırlıkları da dinamik olarak geometrik özelliklere göre belirlenmektedir. Bunlara ek olarak vektör dikkat (vector attention) düzeneği kullanılarak noktaların yönelim bilgisi daha iyi belirlenebilir. Geleneksel skaler dikkat düzeneğinde, sorgu ve anahtar arasındaki ilişki tek bir skaler değer olarak hesaplanır ve bu değer komşu özellikleri modüle eder. Vektör dikkat düzeneği ise bu ilişkiyi her bir özellik kanalı için ayrı ayrı hesaplar ve vektör şeklinde ağırlıklar üretir. Gruplandırılmış vektör dikkati (grouped vector attention) kullanılarak kanal bazında dikkat hesaplamaları optimize edilir ve her kanalın farklı önem düzeylerine göre yeniden ağırlıklandırılması sağlanır. Ardından özellik birleştirme (improved feature aggregation) ile farklı komşuluk kümelerinden elde edilen özellikleri orijinal modele göre daha etkili biçimde birleştirilerek her nokta için daha zengin bir temsil elde edilir. Bu sayede, her bir hiyerarşik katmanda hem yerel detaylar hem de global bağlam daha etkili şekilde modellenmektedir (X. Wu vd., 2022).

PTv2 mimarisinde, PTv1 mimarisindeki gibi öz dikkatin tüm noktalar yerine, nokta bazında dinamik olarak yerel bölgeler üzerinden gerçekleştirilmesi, hesaplama maliyetini ve bellek kullanımını azaltırken, yerel uzamsal bağlamın etkili şekilde modellenmesine olanak tanımaktadır. Mimarideki bu değişiklik, modelin daha büyük ve yoğun nokta bulutlarını daha pratik bir şekilde işlemesini sağlamaktadır. PTv2 modeli, orijinal modele göre büyük ölçekli veri kümelerinde daha hızlı, daha başarılı ve daha verimlidir. Bununla birlikte hâlâ öz dikkat tabanlı olduğu için yoğun verilerde hesaplama maliyeti yani

‘Grafik İşlem Birimi (Graphical Proceecssing Unit, GPU)’ kullanımı yüksektir.

3.3.3 PointTransformer v3 (PTv3) (2023)

PointTransformer v3 mimarisi, PTv2 mimarisinde kullanılan dinamik yapıdaki yerel dikkat düzeneğini, seyreklik, sıklık gibi nokta bulutundaki nokta yoğunluğu değişimlerine daha iyi uyum sağlayabilen uyarlanabilir (adaptive) yarıçaplı dinamik yerel dikkat mekanizmasına dönüştürmüştür. Böylece hem bölütleme başarımı daha iyi duruma gelmiş hem de hesaplama hızı, bellek tüketimi ve hesaplama maliyeti PTv2 modeline göre büyük ölçüde daha iyi hale gelmiştir (Y. Wu vd., 2024).

PointTransformer v3 mimarisine, dikkat düzeneğindeki verimliliği arttırmak ve ölçeklenebilirlik ile hesaplama maliyetleri arasında daha iyi bir denge kurmak için ‘sıralı/serileştirilmiş dikkat (serialized attention)’ ve ‘yama bazlı yerel dikkat (patch attention)’ gibi tamamen farklı dikkat düzenekleri getirilmiştir. Bu sayede, düzensiz ve sırasız nokta bulutları serileştirme teknikleri ile düzenli hale getirilir. Sıralı hale getirilmiş bu dizi, yama bazlı yerel dikkat ile işlenerek yerel özellikleri öğrenmektedir. Böylece, daha geniş bir algılama alanı (receptive field) elde edilirken hesaplama ve bellek kullanımı da azaltılmaktadır. Daha geniş bir algılama alanı uzun mesafeli bağımlılıkları belirlemeyi kolaylaştırmaktadır (Y. Wu vd., 2024).

Sıralı dikkat, nokta bulutunu uzamsal yakınlığa veya özellik benzerliğine göre dizi (sequence) adı verilen küçük alt gruplara bölerek öz dikkat karmaşıklığını düşürür. Space-filling curve, z-order, Hilbert eğrisi gibi belirli bir sıralama tekniği kullanarak nokta bulutunun düzensiz ve sırasız yapısından sıralı bir dizi haline dönüştürülür. Bu sıralama sayesinde, noktalar arasındaki yerel ilişkiler daha düzenli bir biçimde ortaya çıkarılır. Bu serileştirme işlemi, nokta bulutuna yapay bir düzen getirerek, dikkat düzeneğinin bu sıralı yapı üzerinde uygulanmasını sağlar. Böylece, model, nokta bulutunu daha verimli bir şekilde işleyebilir ve dikkat hesaplamalarını düzenli bir veri yapısı üzerinde gerçekleştirebilir. Bu durumda önceki PointTransformer mimarilerinde kullanılan k en yakın komşuluk gibi karmaşık sorgulara ve ofset dikkat düzeneğine gerek kalmaz. Grupların/dizilerin çıktıları, toplu bir şekilde birleştirilerek (serialized) nokta bulutu bütün olarak analiz edilir. Tüm noktaları aynı anda işlenmesi yerine, küçük gruplar halinde işlendiğinden GPU bellek kullanımını azalır (Y. Wu vd., 2024).

Yama bazlı dikkat, nokta bulutunu (serileştirilmiş nokta dizisini) küre veya küp şeklindeki yama adı verilen küçük yerel bölgelere bölmekte ve iki aşamalı dikkat uygulamaktadır. İlki ‘yama içi dikkat (intra-patch attention)’ olup her bir yamada bulunan noktalar arasındaki ilişkiler öğrenilir ve yerel bağlam oluşturulur. Bir duvar parçasının geometrisi buna örnek olarak gösterilebilir. İkincisi ‘yamalar arası dikkat (inter-patch attention)’ olup yamalar arasındaki bağlantılar öğrenilir ve global bağlam oluşturulur. Bir binanın farklı katları buna örnek olarak gösterilebilir. Böylece, model hem ince ayrıntıları hem de genel sahne yapısını öğrenmektedir (Y. Wu vd., 2024).

Sıralı dikkat veri büyüklüğünün yönetilmesini sağlarken, yama bazlı dikkat yerel-global bağlam dengesini sağlamaktadır. PointTransformer v3 mimarisi bu iki düzeneikle genellikle 3B omurga (backbone) olarak kullanılmakta; yani, temel özellik çıkarımını gerçekleştirilmektedir.

PointTransformer v3 mimarisine önemsiz veya gürültülü noktaları elemek için ‘seyrek işleme (sparse processing)’ optimizasyonu eklenmiştir. Bu mimarideki seyrek işleme optimizasyonu sıralı dikkat düzeneğinden önce ‘seyrek evrişim katmanların (sparse convolutional layers)’ kullanımıyla gerçekleştirilir. Seyrek işleme temel olarak girdi nokta bulutunun temel özelliklerini seyrek bir şekilde çıkarmak ve bu özellikleri daha sonraki işlemler için uygun bir forma dönüştürmek için kullanılır. Bu optimizasyon tekniği sayesinde model daha büyük nokta bulutlarını daha az kaynak tüketerek ve daha hızlı biçimde işleyebilmektedir (Graham vd., 2018; Tang vd., 2020).

3.3.4 Point Cloud Transform (PCT) (2020)

Dönüştürücü mimariyi nokta bulutlarına uygulayan diğer öncü bir mimari Point Cloud Transformer (PCT) modelidir. PCT, dikkat düzeneğiyle nokta ilişkilerini öğrenen ve öncelikle global özellikleri öğrenmede etkili olan bir mimariye sahiptir. PCT mimarisinin tasarımı özellikle permütasyon değişmezliğine odaklanmaktadır. Diğer bir ifadeyle, nokta bulutundaki noktaların sırasının; bölütleme, sınıflandırma gibi bilgisayarlı görü görevlerinin sonucunu etkilememesi gerektiği ilkesine özel önem vermektedir (M.-H. Guo vd., 2020).

PCT mimarisinin temel özellikleri şunlardır (M.-H. Guo vd., 2020):

- **Girdi katıştırması (input embedding):** Nokta bulutundaki 3B konum bilgisine sahip her nokta çok katmanlı algılayıcı (MLP) kullanılarak daha yüksek boyutlu bir özellik vektörüne dönüştürülür. Bu, modelin (X,Y,Z) koordinat verilerinden anlamlı özellikler çıkarmasını sağlar.
- Her bir noktayı bir parçacık (token) olarak ele alır, geleneksel dönüştürücü mimarisi ve öz dikkat düzeneğiyle işler.
- **Dönüştürücü katmanları:** PCT, standart dönüştürücü kodlayıcı yapısı kullanır. Her katman, çok başlı öz dikkat (multi-head self-attention) düzeneği ve ileri beslemeli ağ (feed-forward network) içerir. Her bir nokta özellik vektörüne gömüldükten sonra bir öz dikkat düzeneği kullanılarak noktalar arasındaki ilişkiler öğrenilir. Öz dikkat, tüm noktalar arasındaki ikili (pairwise) ilişkileri modelleyerek global bağlamı öğrenir.
- **Konum kodlaması (positional encoding):** Nokta bulutları sırasız bir veri yapısına sahip olduğundan, her noktanın koordinatları konum kodlaması olarak eklenir. Bu, modelin noktaların uzamsal konumlarını anlamasına yardımcı olur. Bu bilgi dikkat düzeneğine entegre edilir.
- Önemsiz noktaları elemek için seyrek işleme tekniği kullanılır. Bu sayede özellikle seyrek verilerde etkili hâle gelir.
- **Özellik çıkarma:** Nokta bulutu özelliklerini öğrenmek için dönüştürücü kodlayıcıları/blokları kullanılarak nokta bulutuna global dikkat düzeneğini uygulanır. Noktalar arasındaki ilişkileri modellemek için öz dikkat düzeneğinden ve ofset düzeneğinden yararlanılır. Bu özellikler, daha sonra sınıflandırma, bölütleme gibi görevler için kullanılır.
- **Dinamik komşuluk belirleme:** Yerel bilgi çıkarımı için en yakın k komşu (kNN) gibi dinamik olarak belirlenen komşuluklar üzerinde öz dikkat düzeneği uygulanır. Böylece her noktanın yerel çevresi tanımlanır ve yerel bağlam bilgisi öğrenilir. Başka bir deyişle, her noktanın özellik vektörü, kendi çevresindeki noktaların özellikleriyle etkileşim içinde yeniden ağırlıklandırılır. Bu sayede, yerel bağlamın önemi korunurken, aynı zamanda global ilişkilerin de modellenmesine olanak tanınmış olunur.
- Çok katmanlı bir yapı ile hiyerarşik özellikler öğrenilir (multi scale learning). Böylece hem nokta bulutunun global yapısı anlaşılmasına çalışılır hem de farklı ölçeklerdeki yerel bilgiler entegre edilebilir. Bu sayede hem global ilişkiler öğrenilirken hem de yerel geometrik ayrıntılar korunmuş olur. Diğer bir ifadeyle yerel özellikler (k-NN ile) ve global özellikler (dönüştürücü kodlayıcıları/katmanları) birleştirilir (local-global fusion). Farklı ölçeklerdeki özellikleri harmanlanarak detay kaybı önlemeye çalışılır.

PCT, dönüştürücü mimarisinin gücünü nokta bulutlarına taşıyarak global bağlamdaki karmaşık ilişkileri öğrenmede etkili bir mimaridir. Küçük ve orta ölçekli nokta bulutlarında eğitim ve çıkarım süreçlerinde verimlidir. Ancak, tüm noktalar arasında ikili (pairwise) etkileşimler hesapladığından ve bu nedenle hesaplama karmaşıklığı arttığından büyük veri kümelerinde hesaplama maliyeti çok yükselmektedir.

3.3.5 Mask3D (2022)

Mask3D, özellikle 3B nokta bulutlarından örnek bölütlemesi maskeleri (instance segmentation mask) tahmin etmek için tasarlanmış dönüştürücü temelli bir mimaridir. Mask3D mimarisi genel olarak 3B özellik çıkarım ağının (3D backbone) ve dönüştürücü tabanlı kod çözücülerin (transformer decoder) bütünleştirilmesinden oluşmaktadır. Nokta bulutundan temel özellikleri çıkarmak için 3B özellik ağı kullanılır ve büyük veri kümelerinde hesaplama verimliliği için seyrek işleme optimizasyonları uygulanır (Schult vd., 2023).

Mask3D mimarisinde her nokta bir örnek sorgu (instance query) olarak temsil edilir. Örnek sorgu, bir sahnedeki her nesneye karşılık gelen öğrenilebilir (trainable) bir temsil vektörüdür. Dönüştürücü tabanlı kod çözücüler (decoders) çok ölçekli nokta bulutu özelliklerini dikkate alarak doğrudan örnek maskelerini tahmin eder. Böylece dönüştürücü tabanlı kod çözücüler hem detaylı yerel yapıları hem de geniş bağlamsal ilişkileri öğrenerek yerel-global dengesini sağlamaktadır. Bunun için her noktaya ilişkin olarak bir sorgu vektörü oluşturulur ve farklı ölçeklerdeki ilişkileri birleştirmek için ‘Çok Başlıklı Öz Dikkat (Multi-Head Attention)’ düzeneği kullanılır. Ardından her sorgu vektörü için nokta bulutunda bir bölütleme maskesi tahmin edilir. Dinamik maske üretim bileşeninde maske sınıflandırması ve uyarlanabilir maske boyutu belirlenir (Schult vd., 2023).

Mask3D mimarisi, dönüştürücü mimarisi ve dinamik maske üretimi biçiminde uyguladığı maske sınıflandırma stratejisini birleştirerek, nokta bulutlarının maske tabanlı tahmini için yenilikçi bir yaklaşım sunmaktadır. Bu yaklaşımla, geleneksel oy verme (voting) ve geometrik kümeleme yöntemlerine duyulan gereksinim azalmakta ve maskenin doğrudan tahmin edilmesi sağlanmaktadır. Bu üstünlük, modelin özellikle karmaşık ve büyük nesnelerin bölütleme görevlerinde detaylı nesne sınırlarının belirlenmesindeki başarımı arttırmaktadır. Ayrıca seyrek veya gürültülü verilerde dahi tutarlı bölütleme görevi gerçekleştirilmesini olanaklı hale getirmektedir.

Mask3D modeli nokta bulutlarındaki nesneleri doğrudan maskeleme işlemi ile bölütleme görevini yerine getirebilmektedir. Bununla birlikte bu modelin kendi mimari bileşenlerinin bütünleştirilmesi gerektiğinden uygulamasında karmaşıklık ortaya çıkabilmektedir.

3.4 Yeni Nesil Nokta Bölütlemesi

Üç Boyutlu Büyük Dil Modeli (3D Large Language Model, 3D-LLM) genel anlamda 3B uzamsal veri kümeleri üzerinde çalışan büyük dil modellerini ifade etmektedir. Diğer bir ifadeyle 3D-LLM; görüntü, video, nokta bulutu, diğer 3B veri kümeleri gibi uzamsal veriyi anlayabilen, yorumlayabilen ve bu konuda sözel (verbal) çıktılar üretebilen uzamsal (coğrafi, mekânsal) dil modelleridir.

3B nokta bulutu, mesh ve voxel gibi 3B uzamsal veri kümeleri ile büyük dil modellerini bütünleştiren 3D-LLM yaklaşımı, 3B sahnelerden ve nokta bulutlarından elde edilen uzamsal, geometrik ve semantik bilgileri doğal dil işleme yetenekleriyle bütünleştirmektedir. Böylece bu yaklaşımla; 3B sahneler hakkında betimlemeler yapılabilir, kullanıcının sorduğu konum bilgisi tabanlı sorular yanıtlanabilir, nesne tespiti ve ayrıntılı 3B sahne analizi yapılabilir ve hatta 3B ortamlarda görev planlaması ya da navigasyon gibi dil tabanlı etkileşimler gerektiren uygulamalar gerçekleştirilebilir.

Nokta bulutlarındaki uzamsal ilişkileri (spatial relationships) modelleyebilecek, nokta bulutunu bölütleme görevlerinde kullanılabilecek ve nokta bulutlarından yapılandırılmış 3B sahneleri üretebilecek 3D-LLM yaklaşımlarına aşağıda kısaca değinilmiştir.

3.4.1 Large Language Model for Spatial Understanding (SpatialLM) (2025)

SpatialLM, Manycore firması tarafından 2025 yılında GitHub ve Hugging Face gibi platformlarda açık kaynak olarak paylaşılmış ve uzamsal anlayış için geliştirilmiş bir LLM olarak tanıtılmıştır. SpatialLM, geleneksel büyük dil modellerinin uzamsal ilişkileri ve gerçek dünya geometrisini anlama konusundaki sınırlamalarını aşmak için tasarlanmış bir dönüştürücü tabanlı derin öğrenme mimarisidir. SpatialLM, 3B uzamsal veri kümelerini bir dil gibi modelleyerek, dil modellerinin uzamsal akıl yürütme yeteneklerinden faydalanmayı amaçlamaktadır. SpatialLM mimarisinin Llama (SpatialLM-Llama-1B) ve Qwen (SpatialLM-Qwen-0.5B) üzerinde modelleri bulunmaktadır (ManyCore Research Team, 2025).

SpatialLM, monoküler video dizileri, RGB-D görüntüler ve LiDAR sensörleri gibi çeşitli kaynaklardan gelen nokta bulutlarını işleyebilen çok modlu bir mimaridir. SpatialLM mimarisi yapılandırılmamış 3B geometrik veriler ile yapılandırılmış 3B temsiller arasındaki boşluğu doldurmakta ve yüksek seviyeli semantik anlayış sunmaktadır. Duvarlar, kapılar, pencereler gibi sahnelerin semantik yapısını dil tabanlı betimlemeler ve etiketlemelerle ifade edebilmektedir (ManyCore Research Team, 2025).

SpatialLM mimarisinin temel bileşenleri aşağıda kısaca ifade edilmiştir (ManyCore Research Team, 2025):

- **Nokta Bulutu Kodlayıcısı (Point Cloud Encoder):** Bu bileşen, girdi olarak verilen 3B nokta bulutu verisini daha kompakt ve anlamlı özellik vektörlerine dönüştürür.
- **Büyük Dil Modeli (Large Language Model, LLM):** Nokta bulutu kodlayıcısından elde edilen nokta bulutu özellikleri, büyük dil modeline girdi olarak verilir. LLM, bu özellikleri analiz ederek sahnenin yapısal ve semantik bir açıklamasını içeren sahne kodlarını üretir. Bu kodlar, duvar, kapı, pencere gibi sahnedeki mimari elemanları ve nesneleri tanımlayan semantik bilgileri içerir.
- **Yapılandırılmış Çıktı Oluşturma:** Üretilen sahne kodları, çeşitli formatlarda yapılandırılmış 3B çıktılara dönüştürülebilir. Bu çıktılar arasında yönlendirilmiş sınırlayıcı kutular veya maskeler, 2B kat planları ve IFC (Industry Foundation Classes) gibi endüstri standardı formatlar yer alabilir. Böylece sahnede bulunan mimari elemanların/nesnelerin konumları, sınırları ve semantik etiketleri çıktı olarak üretilen temsillerde bulunmaktadır.

SpatialLM, temel bir sahne anlayışı ürettiği için, uygun ince ayar (fine-tuning) ve ek segmentasyon başlıkları (segmentation heads) eklenerek nokta bulutları üzerinde anlamsal, örnek ve bütünsel bölütleme görevleri yerine getirilebilir.

3.4.2 PointLLM (2024)

Büyük dil modellerinin, 3B nokta bulutlarını anlayabilmesi ve 3B sahnelerde doğal dil tabanlı analiz yapılabilmesi için geliştirmiş bir modeldir. Böylece 3B veri kümeleri üzerinde doğal dil etkileşimi sağlayarak kullanıcı istemlerini (prompt) anlayabilmekte ve bağlamsal olarak uygun yanıtlar üretebilmektedir (Xu vd., 2023).

PointLLM, nokta bulutundaki her noktanın geometrik ve semantik özelliklerini dil modelleme ile birleştirmek için tasarlanmış bir mimaridir. Geometri, görünüm ve metin kaynaklı bilgileri etkili biçimde bütünleştirmek için bir nokta bulutu kodlayıcısı ve güçlü bir LLM kullanmaktadır. Çapraz dikkat (cross attention) düzeneğiyle nokta bulutu ve metin özellikleri arasındaki ilişkileri kurmaktadır (Xu vd., 2023).

3.4.3 Point-Bind LLM (2023)

Point-Bind LLM, nokta bulutu özelliklerini ImageBind (Girdhar vd., 2023) ile hizalamakta ve çıktı üretimi için ImageBind-LLM gibi 2B büyük dil modellerini kullanmaktadır (Z. Guo vd., 2023). Point-Bind LLM mimarisinin temel bileşenleri

aşağıda kısaca ifade edilmiştir (Z. Guo vd., 2023):

- **Nokta bulutu özelliklerinin çıkarılması:** 3B nokta bulutundan derin öğrenme tabanlı bir model kullanılarak anlamlı özellikler çıkarılır. Bu özellikler, nokta bulutunun geometrik ve yansıma, renk gibi görünüm bilgilerini içerir.
- **ImageBind ile uyumlu hale getirme:** Point-Bind LLM mimarisinin temel yeniliği, çıkarılan nokta bulutu özelliklerini ImageBind adı verilen çok modlu bir katıştırma (embedding) uzayı ile hizalaması yani uyumlu hale getirmesidir. ImageBind, görüntü, metin, ses, video, 3B veri, termal görüntüler vb. farklı modlardaki veriyi ortak bir temsil uzayında bir araya getirebilen bir modeldir (Girdhar vd., 2023). Bu hizalama/uyum sayesinde, nokta bulutu özellikleri, ImageBind modelinin kullanabildiği diğer modlarla (özellikle metin) ilişkilendirilebilir hale gelmiş olur.
- **2B büyük dil modellerinden yararlanma:** Nokta bulutu özellikleri ImageBind uzayına başarıyla getirildikten sonra, bu özellikler ImageBind ile uyumlu olacak şekilde genellikle 2B görüntü ve metin üzerinde eğitilmiş ImageBind-LLM gibi büyük dil modellerini beslemek için kullanılır. Bu sayede, LLM, 3B nokta bulutlarının içeriğini anlayabilir ve bunlarla ilgili metin tabanlı çıktılar üretebilir.

3.5 Nokta Bulutu Bölütleme Yaklaşımlarının Değerlendirilmesi

Yukarıda ifade edilen yaklaşımlardan hangisinin en uygun olacağı; doğruluk, hız, bellek kullanımı, bölütleme ayrıntı düzeyi gibi uygulamanın özel gereksinimlerine, gerçek zamanlı veya sonradan değerlendirme gibi uygulama biçimine, veri kümesinin büyüklüğüne ve kullanılacak hesaplama gücüne bağlıdır. Klasik yöntemler daha basit problemler için yeterli olabilmekte ve daha az hesaplama kaynağı gerektirmektedir. Derin öğrenme tabanlı modern yaklaşımlar ise genellikle daha karmaşık senaryolarda ve büyük veri kümelerinde daha iyi genelleme yapmakta ve daha yüksek başarımlar göstermektedir. Son dönemde dönüştürücü tabanlı mimariler ve çizge yapısına dayalı yöntemler nokta bulutu bölütleme görevlerinde daha fazla ilgi görmektedir. Bununla birlikte dönüştürücü tabanlı mimarilerin hesaplama maliyeti ve veri gereksinimleri hâlâ dikkat edilmesi gereken konular arasında bulunmaktadır.

Nokta bazlı MLP yöntemlerinden olan PointNet modeli öncelikle global özelliklere odaklanmakta ve yerel detayları yeterince temsil edememektedir. PointNet++ ve RandLA-Net modelleri, yerel bağlamı PointNet yöntemine göre daha iyi modellemektedir. RandLA-Net mimarisi, büyük veriler için optimize edilmişken, PointNet++ modeli daha yüksek doğruluk sunmaktadır. PointCNN (Y. Li vd., 2018) ve KPConv gibi evrişim tabanlı yöntemler yerel özellikleri iyi belirlemesine karşın tasarım açısından karmaşıktır. Voxel ve projeksiyon tabanlı yöntemler geleneksel CNN mimarileri ile uyumludur. Voxel tabanlı olan evrişim yöntemleri yoğun bellek kullanmaktadır. Dynamic Graph CNN (DGCNN) (Wang vd., 2018) gibi çizge tabanlı yöntemler yerel ilişkileri modellemede, karmaşık geometrik yapıları çıkarmada başarılıdır ancak hesaplama maliyetleri yüksektir. Dönüştürücü tabanlı mimariler hem yerel hem global özellikleri belirlemede güçlüdür ancak güçlü hesaplama kaynaklarına gereksinim duymaktadır.

PointNet++, PointCNN, KPConv, DGCNN gibi evrişim temelli yaklaşımların farklılaştığı noktalar genel olarak komşu noktaların belirlenmesi, özelliklerin birleştirilmesi ve ağırlıkların/parametrelerin öğrenimidir. Komşuluk seçiminde kNN, yarıçap tabanlı sorgular, kareler ağı tabanlı örnekleme gibi yöntemlerde yararlanılmaktadır. Yerel özelliklerden global özellik temsiline katkıda bulunmak için özellik birleştirme aşamasında havuzlama katmanları veya hiyerarşik yöntemler kullanılmaktadır. Bu tür yöntemlerde genellikle ağırlıklar sabit değildir ve eğitim sırasında öğrenilmektedir. Bu sayede model, veriye özgü yerel geometrik yapıları daha iyi çıkarabilmektedir. Nokta bulutlarına özgü hangi evrişim türünün en

uygun olduğu, uygulamanın gereksinimlerine ve veri kümesinin özelliklerine bağlı bulunmaktadır.

Point Cloud Transform modeli dönüştürücü mimarisi yaklaşımını nokta bulutlarına doğrudan uygulamaya odaklanırken, PointTransformer serisi mimariler yerel komşuluk ilişkilerine daha fazla vurgu yapmaktadır. Bu nedenle PointTransformer serisi mimariler karmaşık ilişkileri modellemede Point Cloud Transform mimarisinden daha iyiyken, Point Cloud Transform mimarisi de yerel ve global özellikleri daha dengeli bir şekilde modelleyebilmektedir. PTv2 modelinin hesaplama verimliliği PCT modelinden daha iyi olarak kabul edilebilir. Point Cloud Transform modeli daha genel bir dönüştürücü mimarisi kullanmakta yani nokta bulutlarını klasik dönüştürücü temelli yaklaşımla doğrudan işlemektedir. Başka bir deyişle PCT modeli nokta bulutlarına uyarlanmış standart bir yaklaşım sunmaktadır. Bu da PCT mimarisinin belirli görevler için PointTransformer kadar optimize edilemeyeceğini göstermektedir. PTv3 mimarisinde ise nokta bulutu yamalara bölünerek çıkarılan özellikler toplu bir şekilde birleştirilmektedir. Bu sayede PTv3 mimarisi özellikle büyük ölçekli verilerde uzun mesafeli bağımlılıkların ortaya çıkarılmasında yani hızlı ve geniş bir bağlamın öğrenilmesinde diğer dönüştürücü mimarilere göre üstünlük kazanmaktadır (H. Zhao vd., 2021; M.-H. Guo vd., 2020; X. Wu vd., 2022; Y. Wu vd., 2024).

PTv3 mimarisi gerek çatlaklar, kırıklar, kolon veya çatı hasarları, deformasyonlar, yıkılmış bölgeler gibi yerel detayların çıkarılmasında gerekse yamaların bütünleştirilmesiyle bina, köprü gibi büyük nesnelerin bütünlüğü analiz etmede kullanılabilecek bir yaklaşımdır. PTv3 mimarisi yerel-global bağlam dengesini iyi biçimde kurabilmesinin yanı sıra deprem sonrası gibi gürültülü veri kümelerinde yamalar arası tutarsızlıkları azaltabilir. Tüm bunlar göz önünde bulundurulduğunda deprem sonrasında nokta bulutlarından bina hasarlarının belirlenmesi görevi için çalışma kapsamında dönüştürücü tabanlı ileri derin öğrenme yaklaşımlarının kullanılması benimsenmiş ve 3B özellik çıkarım ağı olarak PointTransformer v3 mimarisinin kullanılması sonucuna ulaşılmıştır.

Nokta bulutlarının bölütleme görevinde Mask3D mimarisi daha kesin ve net maskeler üretirken, PointTransformer v3 mimarisi daha genel yani ayrıntı düzeyi Mask3D maskelerinden daha düşük bölütleme maskeleri üretebilir. Zaten PointTransformer v3 mimarisinin ana özelliği bölütleme değil 3B özellik çıkarımıdır. Mask3D mimarisi ise deprem sonrasında binalarda oluşacak çatlakların tam sınırlarının belirlenmesi gibi hasar tespitine yönelik daha iyi bölütleme yapılmasına olanak verecektir. Bu nedenle çalışma kapsamında her iki mimarinin birlikte kullanılması önerilmektedir. Diğer bir ifadeyle 3B özellik çıkarıcı olarak PTv3 mimarisi kullanılırken, Mask3D mimarisi de bölütleme başlığı olarak kullanılabilir. Bunun için Mask3D mimarisinde modelin kendinde bulunan 3B özellik çıkarıcının kullanılması yerine PTv3 kullanılacak biçimde değişiklik yapılmalıdır. PTv3 modelinin çıktısının yani ürettiği özelliklerin Mask3D modelinin kod çözücü kısmının beklediği çok ölçekli özellik haritaları formatına ve boyutuna uygun hale getirilmesi gerekmektedir.

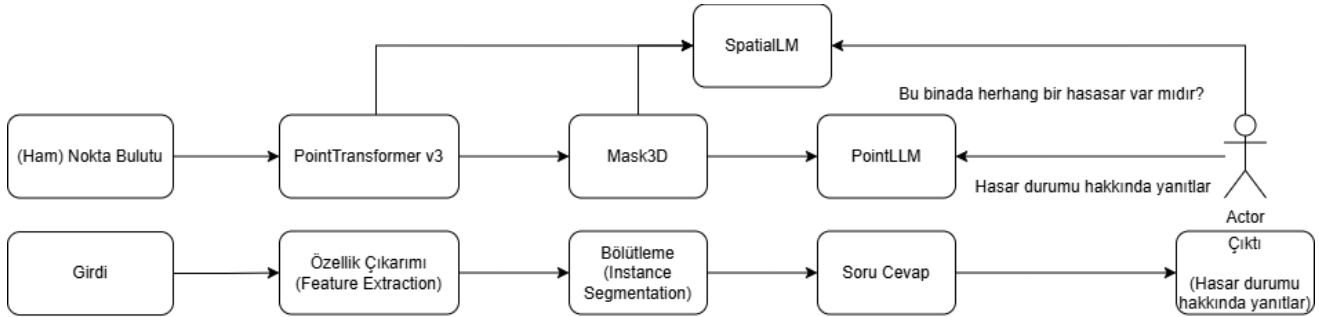
SpatialLM mimarisi, 3B nokta bulutlarından semantik bilgileri çıkarmaya yönelik olarak tasarlanmıştır. SpatialLM ile oluşturulan zengin semantik temsil, deprem sonrası binaların yapısal özelliklerini dilsel betimleme ile ifade edilebilmesine olanak vermektedir.

SpatialLM ile Mask3D modellerinin her ikisi de 3B nokta bulutu veri kümelerini kullansa da SpatialLM modeli daha çok 3B sahne anlayışı, sahnenin semantik betimlemesi ve dil entegrasyonu üzerine odaklanmaktadır. Mask3D modeli ise örnek bölütlemesi üzerine yoğunlaşmakta ve nesne bazlı bölütleme maskesi tahmin işlemini gerçekleştirmektedir. SpatialLM modeli daha çok sahnenin genel düzenini ve mimari unsurlarını anlamaya uygunken, Mask3D daha çok sahnedeki her bir nesne örneği için bölütleme görevini yerine getirmeye çalışmaktadır.

PointLLM modeli nesne seviyesinde anlamaya odaklanırken, SpatialLM daha çok sahne seviyesinde yapısal anlamaya önem vermekte ve yapılandırılmış 3B sahne anlayışı oluşturmada öne çıkmaktadır. PointLLM 3B veri kümeleri üzerinde dilsel

etkileşime girerken, SpatialLM bir sahnenin yapısal ve semantik olarak anlaşılmasına ve yapılandırılmış 3B çıktılar üretilmesine odaklanmaktadır.

Tüm bunlar dikkate alındığında çalışma kapsamında Mask3D bölütleme maskelerinin PointLLM tarafından sorgulanabileceği bir iş hattı (pipeline) oluşturulmuştur. Çalışma kapsamında önerilen bu iş hattına alternatif olarak PTV3 özelliklerinin SpatialLM modelinde anlamlandırılmasına da iş hattı tasarında yer verilmiştir. Şekil 4 çalışma kapsamında tasarlanan iş hattını göstermektedir.



Şekil 4: EQDamage4B_Pnt iş hattının ilk bölümü

4. Modelin Eğitimi ve Veri Kümeleri

Ön eğitim veri kümeleri (pre-training data sets) çoğunlukla genel düzeyde 3B anlama veya bölütleme görevlerine odaklanmaktadır. Deprem sonrası bina hasar tespiti ise genel düzeydeki 3B sahne anlamının çok ötesinde, daha karmaşık bir görevdir. Deprem sonrası nokta bulutlarından hasar belirleme için görüntüler üzerinden hasar tespitinde kullanılan xBD (eXtended Building Damage Dataset) (Gupta vd., 2019; URL-5), gibi kapsamlı, standartlaştırılmış ve toplulukların kullanımına açık bir ön eğitim veri kümesi bulunmamaktadır. Bu nedenle deprem sonrasında nokta bulutlarıyla bina hasar belirleme çalışmalarında genellikle kamu kurumlarının, yerel yönetimlerin ürettiği ve/veya bazı açık kaynaklı LiDAR veri portallarından elde edilebilen nokta bulutları kullanılmaktadır. OpenTopography, USGS Earth Explorer, National Center for Airborne Laser Mapping (NCALM) Data Archive, NASA Disasters Program, IEEE GRSS Data Fusion Contest 2021, DALES ve benzerleri bu tür veri portallarına örnek olarak gösterilebilir. Bu veri kümeleri deprem, sel, yangın gibi doğal afetler sonrasında LiDAR, fotogrametri gibi ölçme teknikleri ile üretilen nokta bulutlarını içermektedir. Bununla birlikte bu tür veri kümeleri genellikle bina hasar etiketleri içermemektedir. Bu durumda, hasarlı ve hasarsız binaların mevcut görüntü verileri ve hasar raporları gibi diğer tür veri kaynakları kullanılarak manuel olarak etiketlenmesi gerekmektedir. Örneğin 6-20 Şubat 2023 Kahramanmaraş ve Hatay depremleri sonrasında üretilen LiDAR ve insansız hava aracı nokta bulutları veri kümeleri, olası Büyük Marmara Depremi için eğitim veri kümesi olarak hazırlanabilir. Böylece olası sonraki depremlerde çatı çökmeleri, zemin deformasyonları vb. bina hasarlarını tespit etmek için sözü edilen etiketlenmiş nokta veri kümeleri kullanılabilir.

Deprem sonrası nokta bulutlarından bina hasar tespiti için kullanılan ve binaların yapısal hasarlarını ayrıntılı bir şekilde analiz etmeye olanak tanıyan Mashiki, Pohang, Haiti gibi LiDAR ölçme tekniğine dayalı veri kümeleri de bulunmaktadır. 2016 yılında Japonya'nın Kumamoto bölgesinde meydana gelen 6.2 büyüklüğündeki depremin ardından oluşturulan Mashiki Town veri kümesi 3B yapısal analiz gerektiren hasar tespiti modelleri için uygundur. Bu veri kümesinde deprem sonrası binaların hasar durumları etiketlenmiştir. Özellikle, nokta bulutlarından binaların çöküp çökmediğini belirlemek için kullanılmış bir veri kümesidir. 2017 yılında Güney Kore'nin Pohang şehrinde meydana gelen 5.4 büyüklüğündeki deprem sonrası oluşturulan Pohang depremi veri kümesi, binaların yapısal hasarlarını analiz etmek için kullanılmıştır. Deprem öncesi

ve sonrası durumları karşılaştırılarak hasar tespiti yapılabilmektedir. 2010 yılında Haiti'de meydana gelen 7.0 büyüklüğündeki depremin ardından oluşturulan Haiti 2010 depremi veri kümesi, binaların çökmüş veya hasarlı olup olmadığını belirlemek için etiketlenmiş veriler içermektedir. Bu veri kümeleri; deprem sonrası hasar tespiti, risk analizi ve iyileştirme çalışmalarında önemli veri kaynağı olarak kullanılmıştır.

Deprem bölgesine ilişkin deprem öncesi ve deprem sonrasına ait nokta bulutu eğitim veri kümeleri bulabilmek olasıdır. Ancak bu tür veri kümeleri genelde hem yaygın değildir hem de büyük ölçekli olmayabilir. Bu tür veri kümeleri mevcut olsa bile bu veri kümelerine erişmek kolay olmayabilir. Veriye erişilebilirlik, depremin gerçekleştiği bölgeye, veri elde etme olanaklarına ve ilgili kurumların veri paylaşım politikalarına bağlıdır. Aynı bölgenin deprem öncesi ve depremin sonrasına ilişkin nokta bulutları elde edilebilirse bu nokta bulutları değişiklik tespiti (change detection) için kullanılabilir. Deprem öncesi ve sonrasına ilişkin nokta bulutları arasında nokta yoğunluğuna, geometrik sapmalara, yükseklik değişimlerine göre fark analizi yapılabilir. Bunun sonucunda bina hasarları belirlenebilir.

Nokta bulutlarında değişiklik tespiti yapmak için aşağıdaki gibi çeşitli teknikler ve yaklaşımlar kullanılabilir:

- **Mesafe Tabanlı Yaklaşımlar:** Deprem öncesi ve sonrasına ait nokta bulutlarına ilişkin birbirine karşılık gelen noktalar veya yüzeyler arasındaki uzaklıklar hesaplanarak deformasyonlar ve çökmeler tespit edilebilir.
- **Yoğunluk Değişim Analizi:** Nokta yoğunluğundaki değişiklikler, örneğin bir binanın çökmesi sonucu oluşan enkaz yığını gibi durumları gösterebilir.
- **Yükseklik Değişim Analizi:** Binaların veya diğer yapıların yüksekliklerindeki azalmalar veya artışlar hasarın bir göstergesi olabilir.
- **Semantik Değişim Analizi:** Eğer nokta bulutlarındaki bina, yol gibi nesneler semantik olarak etiketlenmişse, bir binanın sağlamdan yıkılmışa dönüşmesi gibi deprem öncesi ve sonrası etiketlerdeki değişiklikler hasarı tespit etmek için kullanılabilir.
- **Model Tabanlı Yaklaşımlar:** Deprem öncesi 3B bina modelleri oluşturulup, deprem sonrası nokta bulutlarıyla karşılaştırılarak hasar tespiti yapılabilir.

Nokta bulutlarına dayalı değişiklik tespiti konusu çalışma kapsamında yapılan süreç tasarımı açısından da değerlendirilebilir. Deprem öncesi ve sonrasına ilişkin nokta bulutları üzerinde PointTransformer v3 ve/veya Mask3D ile bölütleme görevi gerçekleştirildikten sonra deprem öncesi ve sonrasına ilişkin bölütleme maskeleri karşılaştırılabilir. Siamese ağları (X. Zhou vd., 2022) kullanılarak deprem öncesine ve sonrasına ait iki nokta bulutu aynı modelde işlenir ve fark vektörleri çıkarılabilir. Bunun için iki ayrı PointTransformer v3 kodlayıcı ile deprem öncesi ve sonrasına ilişkin nokta bulutları işlenir ve ortak bir kod çözücü ile çöken bina kısımları gibi değişim maskeleri üretilebilir. Change3D Model (Y. Wu vd., 2023) gibi bir model kullanılarak sağlam bölgelerin hasarlı bölgelerle karşılaştırılması gibi özellik tabanlı değişiklik tespiti yapılabilir.

Deprem sonrası hasarlı binaların belirlenmesinde kullanılacak etiketli 3B nokta bulutu verisi üretimi emek yoğun ve ekonomik olarak maliyetli olan zorlu bir süreçtir. Diğer taraftan deprem sonrasında bu tür bir veri kümesinin olması hasarlı binaların 3B modellerini veya şehirlerin afet sonrası durumlarını ortaya çıkarabilmek için çok kritiktir. Ayrıca yukarıda ifade edildiği gibi bu konuda standart, isteyen herkesin ulaşabildiği bir veri kümesi bulunmaması önemli bir eksikliktir. Bu nedenle nokta bulutlarıyla bina hasar tespit çalışmalarına ilişkin eğitim ve karşılaştırma (benchmarking) için kullanılacak etiketli veri kümeleri hazırlanmalı ve toplulukların özgürce erişimine sunulması gerekmektedir. 6-20 Şubat 2023 Kahramanmaraş ve Hatay depremleri sonrasında üretilen LiDAR ve insansız hava aracı nokta bulutları üzerinde bu çalışmalar yapılarak açık

kaynaklı 3B afet veri kümesi (3D_Disaster_TR) oluşturulmaya ivedilikle başlanmalıdır. Manuel nokta bulutu etiketleme için CloudCompare, CAD yazılımları veya superglue gibi özel yazılımlar kullanılabilir. Otomatik bölütleme için PointPainer gibi etiketleme araçları kullanılabilir. Bu araçlar kullanılarak nokta bulutları hasarsız, az hasarlı, orta hasarlı, ağır hasarlı, yıkım gibi sınıflarda etiketlenebilir.

4.1 Modelin Eğitim Stratejisi

Çalışma kapsamında nokta bulutlarında özellik çıkarımı için kullanılması önerilen PointTransformer v3 modelinin aşağıda ifade edilen eğitim yaklaşımlarıyla oluşturulacak bir izleme/stratejiyle eğitilmesi gerekmektedir.

4.1.1 Genel Kentsel Sahne Anlayışını Belirleyebilmek İçin Ön Eğitim (General Pre-Training)

Bu yaklaşımda genel amaçlı 3B ön eğitim veri kümeleri kullanılmaktadır. Eğer mevcutsa alanla ilgili (domain specific) ön eğitim veri kümeleri de kullanılabilir.

Bu aşamada PTv3 modelinin genel 3B özellikleri öğrenmesini sağlayacak büyük ölçekli, genel amaçlı çeşitli veri kümelerinden yararlanılabilir. Bu veri kümeleri, yapay öğrenme modelinin binaların ve diğer yapıların genel geometrisi hakkında bilgi edinmesine yardımcı olabilir. Genel amaçlı 3B ön eğitim veri kümeleri açık alan, kapalı alan ve sentetik veri olmak üzere üç farklı türde sınıflandırılabilir. SensatUrban (Hu vd., 2020b; Hu vd., 2022), Semantic3D (Hackel vd., 2017) veya UrbanScene3D (L. Lin vd., 2021) gibi açık alan veri kümeleri binaların dış geometrisi ve bağlamı hakkında bilgi edinmeye yardımcı olurken, ScanNet, ScanNet2000 (Dai vd., 2017), Stanford 3D Indoor Scenes Dataset (S3DIS) gibi kapalı alan veri kümeleri binaların iç yapısı hakkında genel bir anlayış kazanılmasında kullanılabilir. Bunların dışında binaların 3B CAD modellerini içeren ModelNet (URL-6), ShapeNet (URL-7) gibi sentetik veri kümeleri bulunmaktadır.

SensatUrban, Semantic3D (URL-8) veya UrbanScene3D (URL-9) gibi kentsel sahneler içeren bir veri kümesiyle ön eğitim yapılarak, PTv3 modelinin binalar, yollar ve diğer kentsel yapılar arasındaki ilişkileri öğrenmesi sağlanabilir. SensatUrban veri kümesi, büyük ölçeği ve bina odaklı etiketleri nedeniyle bu aşama için en uygun seçenek olarak düşünülebilir. Deprem sonrasında binaların iç yapısında olan çöken tavanlar, ayrı kolonlar, yıkılan duvarlar gibi hasarların belirlenmesinde, bina içi hasar analizlerinin yapılmasında S3DIS, ScanNet, ScanNet2000 vb. veri kümeleri kullanılabilir.

4.1.2 Alana Özgü Görev olan Hasar Tespiti için Özelleştirilmiş Veri Kümeleri Kullanarak İnce Ayar Yapılması (Domain-Specific Pre-training or Fine-Tuning)

Deprem hasar tespiti için özel olarak hazırlanmış etiketli veri olmadan ya da bu tür bir etiketli veri kümesi ile ince ayar yapılmadan, PTv3 modelinden ve üzerine kurulmuş olan Mask3D modelinden yüksek başarımla elde etmek kolay olmayacaktır. Bu nedenle hasar tespiti gibi alana özgü bir görev için, ön eğitim sonrası ince ayar yapılması kritik önem taşımaktadır. Ön eğitim, modelin iyi bir başlangıç noktasına gelmesini sağlar, ancak nihai performans, hedef görev için özel olarak yapılan ince ayar ile elde edilmektedir. Bu nedenle ön eğitimden sonra, modelin deprem sonrası hasar tespiti görevine uyarlanması için hasarlı bina veri kümeleriyle ince ayar yapılması gerekmektedir. Bu aşamada, 6-20 Şubat 2023 Kahramanmaraş ve Hatay depremleri sonrasında üretilecek etiketli nokta bulutu verisi (3D_Disaster_TR) gibi özel bir veri kümesi kullanılabilir.

4.1.3 Gerçek Dünya Verisinin Sınırlı Olduğu Durumlarda Sentetik Hasar Verisi Kullanımı

Çeşitli nesnelerin 3B CAD modellerini içeren ModelNet, ShapeNet gibi veri kümeleri binaların temel şekilleri ve yapısal elemanları hakkında bilgi edinilmesini sağlamaktadır. ModelNet, ShapeNet gibi veri kümeleri her ne kadar sentetik 3B modeller içerse de hasarlı binalarla ilgili model bulundurmamaktadır. Bu nedenle deprem sonrası hasar belirleme için uygun

olmadığı biçiminde değerlendirilebilir. Eğer uygulamada gerçek dünya verisi sınırlıysa, 3B modelleme araçları kullanılarak sentetik olarak hasar görmüş bina modelleri oluşturabilir. PTV3 modeli bu sentetik veri kümeleri ile alana özgü olarak eğitilebilir. Bu yaklaşım hem genel 3B özellikleri hem de hasar örüntülerini öğrenen güçlü bir model oluşturulmasını sağlayabilir. Sentetik hasarlı bina modelleri oluşturup modelleri eğitmek için ModelNet, ShapeNet gibi veri kümelerini kullanmaktansa örneğin UrbanScene3D gibi veri kümeleri ile Blender, Gazebo gibi 3B grafik modelleme yazılımlarında, NVIDIA Omniverse, CARLA ve AirSim gibi benzetim ortamlarında veya Unreal Engine, Unity gibi oyun motorlarında hasarlı bina benzetimleri (simulation) oluşturularak PTV3 modelinin hasarlı yapıları öğrenmesi de sağlanabilir.

4.1.4 Nokta Bulutları için Öz Denetimli Ön Eğitim Yaklaşımı

Uygulamada etiketli nokta bulutu verisi olmadığında veya sınırlı miktarda olduğunda, etiketlenmemiş büyük veri kümelerinde genellikle 'öz denetimli öğrenme (Self-Supervised Learning, SSL)' tekniği kullanılarak anlamlı temsillerin öğrenilebildiği ön eğitim yaklaşımı kullanılabilir. Diğer bir ifadeyle nokta bulutundan genel amaçlı özellik vektörleri çıkarılabilir. Böylece hem veri etiketleme gereksinimi azaltılmış hem de yapay öğrenme modelinin nesne tespiti, bölütleme, sınıflandırma gibi çeşitli 3B bilgisayarlı görü görevleri için daha iyi genelleme yapmasına olanak tanınmış olunur. Bu yaklaşımda kullanılabilecek yöntemler genel olarak aşağıda ifade edilmiştir:

- **Maskleme tabanlı Yöntemler (Masking-based Methods):** Point-BERT, Point-MAE, vb.
- **Yeniden Oluşturma tabanlı Yöntemler (Reconstruction-based Methods):** SONATA, PointNetAE vb.
- **Karşılaştırmalı Öğrenme tabanlı Yöntemler (Contrastive Learning-based Methods):** PointContrast vb.
- **Dönüşüm Tahmini tabanlı Yöntemler (Transformation Prediction-based Methods):** 3D Jigsaw Puzzle, vb.
- **Üretici tabanlı Yöntemler (Generative-based Methods):** PointFlow, PointDif, vb.

Point-BERT, doğal dil işleme alanında geliştirilmiş olan BERT (Bidirectional Encoder Representations from Transformers) mimarisinden esinlenerek nokta bulutları için geliştirilmiş, öz denetimli ve dönüştürücü tabanlı bir öğrenme modelidir. Point-BERT, nokta bulutlarını doğrudan işlemez. Bunun yerine, nokta bulutlarında 'maskeli nokta modelleme (masked point modeling)' tekniğiyle temsiller öğrenmektedir. Bunun için nokta bulutu önce küçük, anlamlı yamalara/öbeklere (point patches) ayrılır. Point-BERT, bu öbekleri rastlantısal olarak maskeler ve bu maskelenmiş öbeklerin özelliklerini tahmin etmeyi öğrenir. Bu, modelin yerel geometrik yapıları ve bağlamı anlamasına yardımcı olur. Her bir öbek, iyi tanımlanmış bir yerel yapıyı temsil eden bir parçacık (token) olarak kabul edilir. Bu parçacıklarla dönüştürücü bir ağ beslenir (Yu vd., 2022).

SONATA (Self-supervised Occlusion-aware TrAnsformer pre-Training for point cloud Autoencoding), nokta bulutu kodlayıcılarının (encoders) ön eğitimi için tasarlanmış bir öz denetimli öğrenme tekniğidir. Özellikle nokta bulutlarındaki örtüşme (occlusion) durumlarını dikkate alarak, oto kodlayıcı (autoencoder) yapısı ile nokta bulutunun temel temsillerini öğrenmeyi hedefler. SONATA tekniğinde, hiyerarşik kod çözümü kullanılmamaktadır (X. Wu vd., 2025).

SONATA, PointTransformer v3 veya başka bir uygun mimari için başlangıç ağırlıkları (pre-trained weights) sağlayabilir. PointTransformer v3 mimarisi, SONATA tekniğinin uygulanabileceği bir omurga ağ (backbone network) olarak kullanılabilir. Başka bir deyişle SONATA tekniğinin öğrenme stratejileri, PTV3 mimarisinin özellik çıkarma yeteneklerinden faydalanabilir. Görüldüğü üzere PointTransformer v3 modeli ve SONATA tekniği birbirini tamamlayabilmektedir. Büyük bir etiketsiz veri kümesi üzerinde SONATA ile ön eğitim yapılmış bir PTV3 modeli, az etiketli veri bulunduran bir veri kümesinde daha iyi performans gösterebilir.

4.2 Modelin Eğitimi Değerlendirmesi

Deprem sonrasında nokta bulutlarına dayalı olarak binaların hasarlı bölgelerinin bölütlenmesi görevinin eğitim aşamasında aşağıdaki iki yaklaşımdan biri uygulanabilir:

- **Yaklaşım 1:** Ön eğitim (pre-training) ve ince ayar (domain spececfic fine-tuning)
 - PointTransformer v3 mimarisinin veya modelinin önce ön eğitimi ardından uygulama alanına özgü veri kümeleri kullanılarak ince ayarı yapılabilir.
- **Yaklaşım 2:** Öz denetimli ön eğitim stratejisi (pre-training strategy) ve ince ayar (domain spececfic fine-tuning)
 - PointTransformer v3 mimarisine veya modeline önce SONATA gibi ön eğitim stratejisi uygulanır, ardından uygulama alanına özgü veri kümeleri kullanılarak ince ayar yapılabilir.

Deprem sonrası bina hasar analizi özel ve karmaşık bir iş tanıımıdır/senaryosudur (use-case). Deprem sonrasında deprem bölgesindeki binalar genellikle düzensiz ve eksik geometriye sahiptir. Doğrudan ön eğitimi yapılmış bir model her ne kadar ortamın genel geometrik ve yapısal özelliklerini/detaylarını öğrense de deprem bölgesindeki nesneler, enkazlar için iyi genelleştirmeler gerçekleştiremeyebilir ve ayrıntılı hasar analizi için yetersiz kalabilir. Bununla birlikte SONATA gibi öz denetimli ön eğitim yaklaşımının uygulanması ise hasarlı yapıların karmaşıklığı için daha uygun olabilir. Çalışma kapsamında gerçekleştirilen süreç tasarımı deprem sonrası bina hasar tespitine ilişkin, özellikle eksik veya parçalanmış bina parçalarının bölütleme görevini daha iyi yerine getirebilmek için öz denetimli ön eğitim yaklaşımlarının uygulanması tercih edilmiştir. Böylece daha başarılı yüksek sonuçlar elde edileceği düşünülmektedir. Diğer taraftan yalnız klasik ön eğitim ve ince ayar yaklaşımı, öz denetimli ön eğitim yaklaşım ve ince ayara göre daha hızlı ve pratik bir iş hattı oluşturacağı açıktır.

Çalışma kapsamında deprem sonrasında nokta bulutlarına dayalı olarak binaların hasarlarının belirlenmesinde diğer bir ifadeyle binaların hasarlı bölgelerinin bölütlenmesi görevinde SONATA ve PointTransformer v3 yaklaşımının birlikte kullanılması önerilmektedir. SONATA gibi bir ön eğitim stratejisi uygulamak, PointTransformer v3 modelinin daha güçlü ve transfer edilebilir özellikler öğrenmesini sağlayacaktır. Bu da deprem sonrası bina hasarı gibi belirli bir alanda ince ayar yapıldığında daha doğru ve güvenilir semantik bölütleme sonuçları elde etme olasılığını artıracaktır. Birinci yaklaşımda ifade edilen temel ön eğitim de faydalı olacaktır, ancak SONATA gibi özel bir strateji muhtemelen daha iyi bir temel sağlayacaktır.

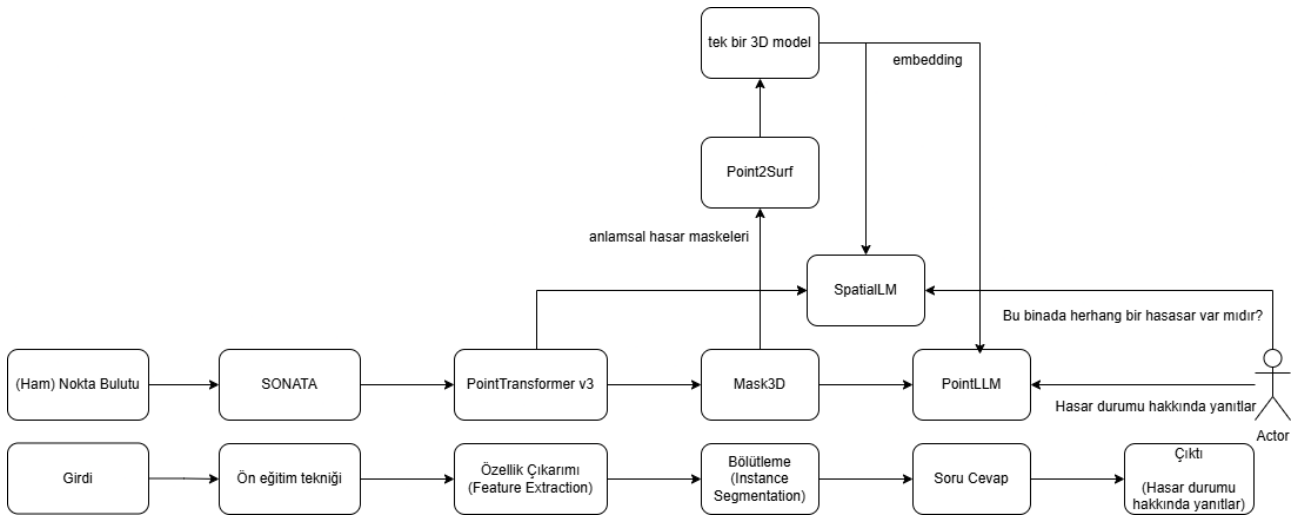
SONATA gibi ön eğitim stratejileri kullanımı, eksik ve bozulmuş bina parçalarını öğrenmeye daha uygun olabilir. Deprem sonrası bina hasar analizi için birbirini kapama/örtme farkındalığına (occlusion-aware) sahip SONATA gibi yöntemler hasar almış, düzensiz bir yapıya dönüşmüş ve eksik geometrisi olan binalar için bina parçalarını öğrenmeye daha uygun olduklarından daha iyi genellemeler yapılmasına yardımcı olabilir. Tam yıkılmış, kısmi yıkılmış, eğik yapılar gibi farklı türdeki hasarları öğrenmek için daha iyi modeller hazırlanmasında destek olabilir. SONATA gibi bir stratejiyle önceden eğitilmiş bir PTV3 modeli, deprem sonrası bina hasarı gibi belirli bir göreve ince ayar yapmaya başladığında daha iyi bir başlangıç noktasına sahip olacaktır. Bu, modelin daha hızlı öğrenmesine, daha az veriyle daha iyi performans göstermesine ve potansiyel olarak daha yüksek doğruluk seviyelerine ulaşmasına olanak tanıyacaktır.

Deprem sonrasında deprem bölgesindeki binalarda eksik veya eğilmiş parçalar olabilir. Veri kümesinde hasarlı, eksik veya örtülü (occluded) bina parçalarının fazla olduğu durumlarda, örtülü alanları modellemeye odaklanan (occlusion-aware) SONATA modeli, Point-BERT modeline göre daha iyi bir seçenek olabilir. Eğer model, bina geometrisini ve yapısal ilişkileri genel olarak öğrenmek için eğitilecekse, bu sefer de Point-BERT modeli daha iyi bir seçenek olabilir. Point-BERT, eksik

parçaları tamamlamaktan çok genel 3B yapı anlayışı geliştirmek için daha kullanışlıdır. Ancak eksik veya bozulmuş bina parçalarını tahmin etmede SONATA kadar güçlü olmayabilir. SONATA yerine Point-BERT kullanılırsa da ikinci yöntem hâlâ en iyi yaklaşım olacaktır ama SONATA kadar örtülü alanları modellemeye odaklanmadığı için eksik parçaları tamamlamada başarımlar düşük kalacaktır.

5. Türkiye'deki Deprem Senaryoları için EQDamage4B_Pnt

Deprem sonrası bina hasar düzeyi belirleme gibi karmaşık uygulamalarda başarımları yüksek sonuçlar elde edebilmek için çalışma kapsamında modüler yapıda bir iş hattı (pipeline) önerilmektedir. Bu iş hattı EQDamage4B_Pnt olarak adlandırılmıştır. EQDamage4B_Pnt etkileşimli çok modlu bir tasarım mimarisidir. Şekil 5'te gösterilen bu iş hattında bazı modeller seçilmiş ve bir iş akışı oluşturulmuştur. 'EQDamage4B_Pnt' ifadesindeki EQ; Earthquake yani deprem, Damage4B; Damage for Building yani bina hasarı, Pnt; Point Cloud yani nokta bulutu anlamında tanımlanmıştır. Bu çalışma kapsamındaki 'EQDamage4B_Pnt' tasarımı daha büyük bir ailenin tek bir bileşenidir. Örneğin aynı görev uydur görüntüleri dayalı gerçekleştirilecekse 'EQDamage4B_Sat' veya aynı görev radar görüntüleri için sel felaketi için gerçekleştirilecekse de 'FLDamage4B_Sar' biçiminde adlandırılmaktadır. Burada FL; Flood yani sel, Sar; Synthetic Aperture Radar yani Yapay Açıklıklı Radar, Sat; Satellite Imagery yani Uydur Görüntüsü yerine kullanılmaktadır.



Şekil 5: EQDamage4B_Pnt iş hattının genel görünümü

Önerilen iş hattında PointTransformer v3 (PTv3) modeli nokta bulutundan özellikleri çıkarmak için kullanılmıştır. PTv3 modeli ile çıkarılacak yüksek kaliteli özellikler, sonraki aşamada bölütleme görevinin başarılı biçimde gerçekleştirilmesi için kullanılmıştır. Bu iş hattında örnek bölütleme görevi için Mask3D seçilmiştir. PTv3 modeli kullanılarak çıkarılan zengin özellikler; Mask3D bölütleme başlığına aktarılmış, daha doğru ve ayrıntılı bölütleme sonuçları elde edilmiştir.

Deprem sonrası bina hasar düzeyi tespiti gibi bir uygulamada, hem bina unsurlarının detaylı bölütleme görevi hem de bölütleme sonucunda elde edilen bölütleme maskelerine ilişkin dilsel analiz/etkileşim önemli rol oynamaktadır. 3B sahne analizinden elde edilen ayrıntılı veri kümeleriyle zengin ve bağlama duyarlı yanıtlar üretmek için dilsel/metinsel etkileşimden yararlanılabilir.

Mask3D den elde edilen sonuçlar PointLLM için girdi olarak değerlendirilmiştir. Bu sayede PointLLM, bu bölütleme sonuçlarını girdi olarak kullanarak 3B sahnenin semantik olarak yorumlayabilmekte ve binaların hasar durumu hakkındaki soruları cevaplayabilmektedir.

Bu iş hattı; SONATA modelinin ön değerlendirmedeki üstünlüğünü, PointTransformer v3 modelinin güçlü özellik çıkarımını, Mask3D modelinin ayrıntılı 3B bölütleme yeteneklerini ve PointLLM modelinin dil tabanlı etkileşim gücünü bir araya getirerek, deprem sonrası hızlı hasar değerlendirmesi ve müdahale planlaması için kapsamlı bir çözüm sunmaya çalışmaktadır. Böylece bina hasar düzeyi belirlenmesi temelinde acil durum müdahalesi için planlama yapılabilir, afet bölgesindeki ve enkazdaki nesneler hızlıca sorgulanabilir.

Önerilen iş hattı ile yanıt verilebilecek sorulara örnek olarak aşağıdakiler gösterilebilir:

- Deprem bölgesinde kaç tane bina hasar görmüştür? Konumları nedir?
- Hasar gören binalardan kaçının çatısı çökmüştür?
 - Örnek Yanıt 1: Bina A (çatı çökmüş), Bina B (sağlam), ...
 - Örnek Yanıt 2: Çatısı çöken binalar: Bina A. Koordinatlar: (X, Y, Z).
- Bu binanın çatısındaki hasar oranı nedir?
 - Örnek Yanıt: Çatısı çöken Bina A'nın çatısındaki hasar oranı %25.
- Deprem bölgesinde hangi alanlarda ciddi hasar vardır?
- Seçilen binanın hasar durumu nedir?
 - Örnek Yanıt: Bina A'nın kuzey cephesinde %40 hasar tespit edilmiştir.
- Koordinatları $36^{\circ} 12' 23.58''$ N ve $36^{\circ} 9' 25.992''$ E olan binanın hasar durumu nedir? Binanın yapı malzemeleri nelerdir? Hasarlı kolonları hangileridir?

Önerilen EQDamage4B_Pnt iş hattı, deprem sonrası hasar tespitinin gerektirdiği yüksek ayrıntı ve çok ölçekli bilgi gereksinimini karşılamak adına stratejik olarak seçilen mimari bileşenlerin sentezinden oluşmaktadır. Ön eğitim aşamasında, deprem enkazındaki düzensiz, parçalanmış ve örtülü (occluded) yapıları öğrenmede genel bir 3B yapı anlayışı sunan Point-BERT gibi yöntemler yerine, örtüşme farkındalığına (occlusion-aware) sahip SONATA tekniği tercih edilerek hasarlı bina parçalarını ve eksik geometrileri öğrenme kabiliyeti arttırılmıştır. Özellik çıkarıcı (backbone) katmanında, yerel bağlamı yakalamakta zorlanan PointNet veya yüksek hesaplama maliyetine sahip DGCNN ve KPConv gibi yöntemler yerine PointTransformer v3 (PTv3) benimsenmiştir. PTv3, sıralı dikkat (serialized attention) düzeneği sayesinde büyük ölçekli ve gürültülü veri kümelerinde düşük bellek kullanımı ve yüksek hız sunarak yerel-global bağlam dengesini etkili bir şekilde kurmaktadır. Bölütleme aşamasında ise geleneksel yöntemlerin aksine Mask3D mimarisi kullanılarak nesneler doğrudan maskeleme yoluyla bölütlenmiş, böylece hasar tespitinde kritik olan çatlak veya kolon sınırları gibi alanlarda daha kesin sonuçların elde edilmesi hedeflenmiştir. Sürecin son halkası olarak entegre edilen PointLLM ve SpatialLM katmanları, elde edilen teknik verileri doğal dil ile bütünleştirerek afet sonrası müdahale ekipleri için hasar durumunun doğrudan sorgulanabildiği etkileşimli ve operasyonel bir karar destek mekanizması olarak iş hattına eklenmiştir. Çalışma kapsamında önerilen süreç tasarımı seçilen modellerin alternatif olabilecek diğer modellerle karşılaştırılması Tablo 1'de yapılmıştır.

Önerilen bu iş hattının deprem sonrası hasar belirleme gibi karmaşık bir görevde başarılı biçimde kullanılabilmesi için her üç modelin de uygun ve doğru etiketlenmiş veri kümeleri ile eğitilmesi ve ince ayarlarının yapılması gerekmektedir. Aksi takdirde genel amaçlı bu modeller, deprem sonrası hasara ilişkin özelleştirme yapılmadan yetersiz kalacaktır. Bununla birlikte ortaya çıkacak bazı diğer zorlukların da çözülmesi gerekmektedir. Bunlardan ilki modellerin bütünleştirilmesi yani modellerin uçtan uca birleştirilmesidir. PointTransformer v3, Mask3D ve PointLLM modellerinin birbiriyle uyumlu çalışması

için girdi ve çıktı formatlarının uygun/uyumlu hale getirilmesi gerekmektedir. Örneğin PTv3 modelinin ürettiği özelliklerin, Mask3D bölütleme başlığına uygun formatta aktarılması gerekmektedir.

Tablo 1: EQDamage4B_Pnt bileşenleri ve alternatif derin öğrenme yaklaşımlarının karşılaştırılması

Bileşen	Alternatif yöntemler	EQDamage4B_Pnt	Neden EQDamage4B_Pnt?
Ön Eğitim	Point-BERT, PointMAE	SONATA	Enkaz gibi örtülü (occluded) ve eksik verilerde yüksek dayanıklılık
Backbone	PointNet++, DGCNN, KPConv	PTv3	Sıralı dikkat ile büyük ölçekli veride düşük bellek kullanımı ve yüksek hız
Bölütleme	Klasik U-Net Başlığı, Voting	Mask3D	Karmaşık nesne sınırlarında doğrudan maske tahmini ile daha yüksek hassasiyet
Analiz	Geometrik/Sınıf Bazlı Analiz	3D-LLM	Hasar durumunun doğal dilde sorgulanabilir ve yorumlanabilir olması
Nokta bulutu girişi	Voxelization, Mesh-first (MeshCNN), SDF/implicit surfaces	Raw point cloud (no early discretization)	Erken voxel/mesh dönüşümü bilgi kaybı ve çözünürlük bağımlılığı yaratır; ham nokta bulutu deprem sonrası düzensiz ve eksik veriler için daha uygundur
Özellik çıkarımı	PointNet++, KPConv, SparseConvNet	PointTransformer v3	Global self-attention ile uzun menzilli ilişkileri yakalar; hasar desenleri lokal değil bağlamsaldır
Ölçeklenebilirlik	Dense CNN tabanlı 3D ağlar	Window-based attention (PTv3)	Büyük ölçekli kentsel sahnelerde bellek ve hesaplama açısından daha ölçeklenebilirdir
Instance segmentation	Semantic-only segmentation, heuristic clustering	Mask3D	Yapı bazlı hasar analizinde instance-level ayırım zorunludur; semantic segmentation yeterli değildir
Geometri–semantik ayrımı	Joint semantic–geometry ağları	Geometri + instance ayrımı sonrası semantik bağlama	Önce geometrik ayrışma, sonra semantik yorumlama; hata yayılımını azaltır
Semantik temsil	Klasik label setleri, rule-based tagging	PointLLM	Sabit sınıflar yerine açık uçlu, açıklanabilir ve genişletilebilir semantik ifade
Uzamsal akıl yürütme	GIS tabanlı sorgular, post-processing kuralları	SpatialLLM	Hasar–komşuluk–yapısal ilişki sorgularını doğal dil ve sembolik akıl yürütme ile birleştirir
Sorgulanabilirlik	SQL/GIS query, API tabanlı filtreleme	LLM tabanlı doğal dil sorguları	Uzman olmayan kullanıcılar için erişilebilirlik; afet sonrası hızlı analiz
İnsan–AI etkileşimi	Statik çıktı raporları	Interactive LLM-based reasoning	Analist geri beslemesi ve senaryo bazlı keşif mümkün
Yüzey üretimi	Poisson reconstruction, marching cubes	Point2Surf (semantic-aware)	Semantik olarak zenginleştirilmiş yüzey üretimi; hasar görselleştirmesi için daha anlamlı

PointLLM modelinin cevaplayabileceği soruların derinliği, Mask3D modelinin bölütleme sonuçlarındaki hasar sınıflandırma ayrıntı düzeyine bağlı olacaktır. Bu nedenle Mask3D modelinde hasar ayrıntı düzeyi olabildiğince yüksek olmalı yani hasarsız, az hasarlı, orta hasarlı, ağır hasarlı, yıkım gibi 5 sınıfta etiketleme çalışması yapılmalıdır.

Yapay öğrenme modellerinin kent yapılarını, kent sahnelerini ve bina geometrilerini öğrenebilmesi için SensatUrban veya Semantic3D gibi geniş ölçekli açık alan veri kümeleri kullanılabilir. Bina içi hasar durumlarının tespiti için ise ScanNet veya S3DIS gibi veri kümeleri kullanılabilir. Bu açık veri kümeleriyle ön eğitim tamamlanabilir. Bununla birlikte Türkiye'deki depremlerin Türkiye yapı stokuna olan etkiyi doğru biçimde değerlendirebilmek için 6-20 Şubat 2023 Kahramanmaraş ve Hatay depremleri sonrasında üretilen LiDAR ve insansız hava aracı nokta bulutları veri kümeleri kullanılarak modellerin ince ayarları yapılması gerekmektedir. Böylece modelin çatlaklar, çökmeler gibi bina hasar örüntülerini ayırt edebilme yeteneği iyileştirilmiş olacaktır.

Önerilen 3 ana aşamalı iş hattının amaca uygun kullanımı yüksek hesaplama maliyetini karşılayacak türde bir hesaplama gücü gerektirecektir. Özellikle bu iş hattının gerçek zamanlı kullanımı düşünülüyorsa bu iş hattının baştan sona hız ve verimlilik açısından optimize edilmesi gerekmektedir. Bunun için çalışmada kullanılacak hesaplama birimi eğitim (training) ve çıkarım (inference) iki farklı biçimde ele alınmalıdır. Aslında bu yaklaşıma dayalı çözüm daha profesyonel olarak ele alınacaksa üçüncü bir aşama da ince ayar (fine-tuning) ve RAG (Retrieval Augmented Generation) için farklı donanım seçimidir. Bütçe kısıtı olması durumunda ince ayar ve RAG içinde eğitim aşamasındaki donanım kullanılabilir. Hesaplama

(compute) bileşeni her ne kadar AI hesaplamalarında GPU olarak öne çıksa da CPU, RAM, NPU birimleriyle birlikte depolama ve ağ katmanlarını da dikkate almak gerekmektedir. Bu akış açısından model ve hesaplama ilişkisi GPU, RAM, depolama üzerinden Tablo 2’de verilmiştir.

Tablo 2: Sürecin gerçekleştirilmesi için gerekli bilgi işlem gücü

Model	Operasyonel aşama	İşlevi	GPU	Bellek (VRAM)	Bellek (RAM)	Depolama (Disk)	Tahmini eğitim süresi
SONATA (Öz denetimli ön eğitim)	Eğitim (Training)	Öz-denetimli Ön Eğitim	NVIDIA A100 / H100 veya 4-8x RTX	40GB+	24GB+	100-500 GB	3-7 Gün (A100 x4)
PT v3	Eğitim, İnce Ayar & Çıkarım	Özellik Çıkarıcı (Backbone)	NVIDIA RTX 3090 / 4090	24GB+	32GB+	100GB (Ağırlıklar: ~1GB)	Scannet veri kümesi ile tek GPU ile 2-4 gün
Mask3D (Örnek Bölütlemesi)	İnce Ayar (Fine-Tuning) & Çıkarım	Örnek Bölütleme Başlığı	NVIDIA A100	40GB+	48GB+	100-300 GB	Scannet veri kümesi ile tek GPU veya 2 GPU ile 3-6 gün
PointLLM (3D LLM Etkileşim)	İnce Ayar, RAG & Çıkarım	Dilsel Etkileşim (Llama-tabanlı)	NVIDIA A100	80GB+	48GB+	150GB+	1-2 gün (fine-tuning)
SpatialLLM	İnce Ayar, RAG & Çıkarım	Uzamsal Anlayış (Qwen-tabanlı)	NVIDIA RTX 4090	80GB+	48GB+	50GB+	1-2 gün (fine-tuning)
Point2Surf (Yüzey yeniden yapılandırma)	Çıkarım (Inference)	Yüzey Rekonstrüksiyonu	NVIDIA RTX 3080+	12GB+	24GB+	50GB+	Saatler, maks. 1 gün

6. Sonuçlar

Çalışma kapsamında deprem sonrasındaki büyük ve karmaşık nokta bulutlarının bölütlenmesi görevinde hem modelin temsil gücünü hem de hesaplama verimliliğini arttıracak bir iş hattı tasarımı yapılmıştır. EQDamage4B_Pnt çatkısı olarak isimlendirilen bu iş hattı Şekil 5’te verilmiştir. Deprem sonrası bina hasar belirleme görevinde PointTransformer v3 modeli hem dinamik yerel detayları/özellikleri çıkarırken hem de global bağlam öğrenmesi sayesinde genel sahne özelliklerini çıkarabilmektedir. Benzer biçimde Mask3D modeli de çatlaklar gibi hem yerel detayları hem de binanın bütünü gibi global maskeleri üretebilmektedir. Bu nedenle çalışma kapsamında önerilen mimaride her iki modelin bütünsel çalışmasının tasarımı yüksek başarılı bir bölütleme görevinin gerçekleştirilmesini olanaklı kılmaktadır. Tasarlanan mimarinin ve/veya kullanılan yapay öğrenme modelinin deprem sonrası bina hasar tespiti görevindeki başarımı, ilgili veri kümeleriyle nasıl bir eğitim stratejisiyle eğitildiğiyle doğrudan ilişkilidir. Bu nedenle çalışma kapsamında süreç modellemesinde ön eğitim stratejileri de dikkate alınmış ve SONATA öz denetim ön eğitim modeli tasarıma dahil edilmiştir. Bölütleme maskeleriyle dilsel etkileşim kurabilmek için PointLLM modeli tasarımda kullanılmıştır. PointLLM, bölütleme sonuçlarını girdi olarak kullanarak 3B sahnenin semantik olarak yorumlayabilmekte ve binaların hasar durumu hakkındaki soruları cevaplayabilmektedir. Çalışma kapsamında üretilen 3B modelin yüzey yeniden yapılandırması için de Point2Surf modeli tasarıma dahil edilmiştir.

Deprem sonrasında kentteki binaların nokta bulutlarına dayalı olarak hasar durumlarının belirlenmesi için yapay öğrenme modellerinin eğitiminde kullanılacak etiketlenmiş, açık ve özgür bir eğitim veri kümesinin toplumun ve geliştirici toplulukların kullanımına ivedilikle açılmalıdır. 6-20 Şubat 2023 Kahramanmaraş ve Hatay depremleri sonrasında üretilen LiDAR ve insansız hava aracı nokta bulutları üzerinde bu çalışmalar yapılarak açık kaynaklı 3B afet veri kümesi (3D_Disaster_TR) hızlıca oluşturulmaya başlanmalıdır.

Veri kümeleri hazırlandıktan sonra çalışma kapsamında önerilen mimarilerin ve modellerin çalıştırılmasını sağlayacak

hesaplama altyapısının oluşturulması ve geliştiricilerin/toplulukların bu altyapıya erişimlerinin sağlanması bu tür dönüştürücü tabanlı yapay öğrenme modellerinin gerçek dünya problemlerinin çözümünde kullanılması açısından kritik öneme sahiptir.

Çıkar Çatışması Beyanı

Yazar, bu çalışmada bilinen ilgili herhangi bir finansal veya finansal olmayan çıkar çatışması olmadığını beyan eder.

Kaynaklar

- Achlioptas, P., Diamanti, O., Mitliagkas, I., & Guibas, L. (2017). Learning Representations and Generative Models for 3D Point Clouds. *35th International Conference on Machine Learning, ICML 2018, 1*, 67-85. <https://arxiv.org/pdf/1707.02392>.
- Chen, C., Wang, B., Lu, C. X., Trigoni, N., & Markham, A. (2023). Deep learning for visual localization and mapping: A survey. *IEEE Transactions on Neural Networks and Learning Systems*, 35(12), 17000-17020.
- Chen, C., Wu, Y., Dai, Q., Zhou, H.-Y., Xu, M., Yang, S., Han, X., & Yu, Y. (2024). A survey on graph neural networks and graph transformers in computer vision: A task-oriented perspective. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12), 10297-10318.
- Dai, A. and C., Angel X. and Savva, Manolis and Halber, Maciej and Funkhouser, & Thomas and Niessner, M. (2017). *ScanNet: Richly-annotated 3D Reconstructions of Indoor Scenes*. <http://www.scan-net.org/>.
- Duan, Y., Zheng, Y., Lu, J., Zhou, J., & Tian, Q. (2019, Haziran). Structural relational reasoning of point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* içinde (s. 949-958). Long Beach, California, ABD.
- Frome, A., Huber, D., Kolluri, R., Bülow, T., & Malik, J. (2004). Recognizing objects in range data using regional point descriptors. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 3023, 224-237.
- Girdhar, R., El-Nouby, A., Liu, Z., Singh, M., Alwala, K. V., Joulin, A., & Misra, I. (2023, Haziran). ImageBind: One Embedding Space To Bind Them All. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 15180-15190). Vancouver, Kanada.
- Graham, B., Engelcke, M., & van der Maaten, L. (2018, Haziran). 3D Semantic Segmentation with Submanifold Sparse Convolutional Networks. *CVPR 2018* içinde (s. 9224-9232). Salt Lake City, Utah, ABD.
- Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). *Deep learning*, 1(2). Cambridge: MIT press.
- Guo, M.-H., Cai, J.-X., Liu, Z.-N., Mu, T.-J., Martin, R. R., & Hu, S.-M. (2020). PCT: Point cloud transformer. *Computational Visual Media*, 7(2), 187-199.
- Guo, Z., Zhang, R., Zhu, X., Tang, Y., Ma, X., Han, J., Chen, K., Gao, P., Li, X., Li, H., & Heng, P. A. (2023). Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*.
- Gupta, R., Hosfelt, R., Sajeev, S., Patel, N., Goodman, B., Doshi, J., Heim, E., Choset, H., & Gaston, M. (2019). xbd: A dataset for assessing building damage from satellite imagery. *arXiv preprint arXiv:1911.09296*.
- Hackel, T., Savinov, N., Ladicky, L., Wegner, J. D., Schindler, K., & Pollefeys, M. (2017). Semantic3D.net: A new Large-scale Point Cloud Classification Benchmark. *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, 4(1W1), 91-98.
- He, Y., Yu, H., Liu, X., Yang, Z., Sun, W., Anwar, S., & Mian, A. (2021). Deep learning based 3D segmentation: A survey. *arXiv preprint arXiv:2103.05423*.
- Hoang, L., Lee, S. H., Kwon, O. H., & Kwon, K. R. (2019). A deep learning method for 3D object classification using the wave kernel signature and a center point of the 3D-triangle mesh. *Electronics*, 8(10), 1196.
- Hoppe, H., DeRose, T., Duchamp, T., McDonald, J., & Stuetzle, W. (1992, Temmuz). Surface reconstruction from unorganized points. *Proceedings of the 19th annual conference on computer graphics and interactive techniques* içinde (s. 71-78). Los Angeles, California,

ABD.

- Hu, Q., Yang, B., Xie, L., Rosa, S., Guo, Y., Wang, Z., Trigoni, N., & Markham, A. (2020a, Haziran). RandLA-Net: Efficient semantic segmentation of large-scale point clouds. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition* içinde (s. 11108–11117).
- Hu, Q., Yang, B., Khalid, S., Xiao, W., Trigoni, N., & Markham, A. (2020b, Haziran). Towards Semantic Segmentation of Urban-Scale 3D Point Clouds: A Dataset, Benchmarks and Challenges. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 4975-4985).
- Hu, Q., Yang, B., Khalid, S., Xiao, W., Trigoni, N., & Markham, A. (2022). SensatUrban: Learning Semantics from Urban-Scale Photogrammetric Point Clouds. *International Journal of Computer Vision*, 130(2), 316-343.
- Johnson, A. E., & Hebert, M. (1999). Using spin images for efficient object recognition in cluttered 3D scenes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5), 433-449.
- Khoshelham, K., & Elberink, S. O. (2012). Accuracy and resolution of kinect depth data for indoor mapping applications. *Sensors*, 12(2), 1437-1454.
- Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6), 84-90.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2323.
- Li, J., Chen, B. M., & Lee, G. H. (2018, Haziran). SO-Net: Self-Organizing Network for Point Cloud Analysis. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 9397-9406). Salt Lake City, Utah, ABD.
- Li, Y., Bu, R., Sun, M., Wu, W., Di, X., & Chen, B. (2018). PointCNN: Convolution on X-transformed points. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 820–830.
- Lin, L., Liu, Y., Hu, Y., Yan, X., Xie, K., & Huang, H. (2021). Urbanscene3d: A large scale urban scene dataset and simulator. *arXiv preprint arXiv:2107.04286*, 2(3).
- Lin, Z., He, L., Yang, H., Sun, X., Zhang, G., Chen, W., Guan, Y., & Zhang, H. (2024, Ekim). SWCF-Net: similarity-weighted convolution and local-global fusion for efficient large-scale point cloud semantic segmentation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* içinde (s. 2602-2609). Abu Dhabi, BAE.
- Liu, W., Sun, J., Li, W., Hu, T., & Wang, P. (2019). Deep learning on point clouds and its application: A survey. *Sensors*, 19(19), 4188.
- Lu, D., Xie, Q., Wei, M., Gao, K., Xu, L., & Li, J. (2022, Haziran). Transformers in 3D Point Clouds: A Survey. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 5560-5568). New Orleans, Louisiana, ABD.
- ManyCore Research Team. (2025). *SpatialLM: Large Language Model for Spatial Understanding*. <https://manycore-research.github.io/SpatialLM/>.
- Qi, C. R., Su, H., Mo, K., & Guibas, L. J. (2017a, Temmuz). PointNet: Deep learning on point sets for 3D classification and segmentation. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* içinde (s. 652–660). Honolulu, Hawaii, ABD.
- Qi, C. R., Yi, L., Su, H., & Guibas, L. J. (2017b). Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *Advances in neural information processing systems*, 30.
- Quan, S., Yu, J., Nie, Z., Wang, M., Feng, S., An, P., & Yang, J. (2024). Deep learning for 3d point cloud enhancement: A survey. *arXiv preprint arXiv:2411.00857*.
- Rusu, R. B., Blodow, N., & Beetz, M. (2009, Mayıs). Fast point feature histograms (FPFH) for 3D registration. *2009 IEEE international conference on robotics and automation* içinde (s. 3212-3217). Kobe, Japonya.
- Sabour, S., Frosst, N., & Hinton, G. E. (2017). Dynamic routing between capsules. *Advances in neural information processing systems*, 30.
- Schult, J., Engelmann, F., Hermans, A., Litany, O., Tang, S., & Leibe, B. (2023, Mayıs). Mask3D: Mask Transformer for 3D Semantic Instance Segmentation. *Proceedings - IEEE International Conference on Robotics and Automation* içinde (s. 8216-8223). Londra, Birleşik Krallık.
- Tang, H., Liu, Z., Zhao, S., Lin, Y., Lin, J., Wang, H., & Han, S. (2020, Ağustos). Searching efficient 3d architectures with sparse point-voxel convolution. *European conference on computer vision* içinde (s. 685-702).
- Thomas, H., Qi, C. R., Deschaud, J. E., Marcotegui, B., Goulette, F., & Guibas, L. (2019, Ekim). KPConv: Flexible and Deformable

- Convolution for Point Clouds. *Proceedings of the IEEE International Conference on Computer Vision* içinde (s. 6410-6419). Seul, Kore.
- Tombari, F., Salti, S., & Di Stefano, L. (2010). Unique signatures of histograms for local surface description. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 6313 LNCS(PART 3), 356-369.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is All You Need. *Advances in Neural Information Processing Systems (NeurIPS 2017)*, 30.
- Wang, Y., Sun, Y., Liu, Z., Sarma, S. E., Bronstein, M. M., & Solomon, J. M. (2018). Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics*, 38(5), 13.
- Wehr, A., & Lohr, U. (1999). Airborne laser scanning—an introduction and overview. *ISPRS Journal of Photogrammetry and Remote Sensing*, 54(2-3), 68-82.
- Wijaya, K. T., Paek, D. H., & Kong, S. H. (2024). Advanced feature learning on point clouds using multi-resolution features and learnable pooling. *Remote Sensing*, 16(11), 1835.
- Wu, X., Lao, Y., Jiang, L., Liu, X., & Zhao, H. (2022). Point Transformer V2: Grouped Vector Attention and Partition-based Pooling. *Advances in Neural Information Processing Systems*, 35. <https://arxiv.org/pdf/2210.05666>.
- Wu, X., DeTone, D., Frost, D., Shen, T., Xie, C., Yang, N., Engel, J., Newcombe, R., Zhao, H., & Straub, J. (2025). *Sonata: Self-Supervised Learning of Reliable Point Representations*. <https://arxiv.org/pdf/2503.16429>.
- Wu, Y., Gu, Y., Chen, X., Liu, H., Zhuang, Y., & Ma, Y. (2023, Haziran). Change3D: A dataset and benchmark for 3D change detection. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* içinde (s. 15254–15264). Vancouver, Kanada.
- Wu, Y. Q., Chen, H. X., & Zhang, Y. (2024). Review of 3D point cloud processing methods based on deep learning. *Chinese Journal of Lasers*, 51(5), 0509001.
- Xu, R., Wang, X., Wang, T., Chen, Y., Pang, J., & Lin, D. (2023). *PointLLM: Empowering Large Language Models to Understand Point Clouds*. <https://arxiv.org/pdf/2308.16911>.
- Yu, X., Tang, L., Rao, Y., Huang, T., Zhou, J., & Lu, J. (2022, Haziran). Point-BERT: Pre-training 3D Point Cloud Transformers with Masked Point Modeling. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 19291-19300). New Orleans, Louisiana, ABD.
- Zaheer, M., Kottur, S., Ravanbakhsh, S., Poczos, B., Salakhutdinov, R. R., & Smola, A. J. (2017). Deep sets. *Advances in neural information processing systems*, 30.
- Zhao, H., Jiang, L., Fu, C. W., & Jia, J. (2019, Haziran). Pointweb: Enhancing local neighborhood features for point cloud processing. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 5560-5568). Long Beach, California, ABD.
- Zhao, H., Jiang, L., Jia, J., Torr, P. H. S., & Koltun, V. (2021, Haziran). Point Transformer. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* içinde (s. 16259-16268). Montreal, Québec, Kanada.
- Zhao, Y., Birdal, T., Deng, H., & Tombari, F. (2019, Haziran). 3D Point-Capsule Networks. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2019-June*, içinde (s. 1009-1018). Long Beach, California, ABD.
- Zhou, X., Qi, X., & Xu, Y. (2022). 3DSiamese: A Siamese network for 3D change detection in point clouds. arXiv. <https://arxiv.org/abs/2203.15227>.
- URL-1: *LASer (LAS) File Format Exchange Activities – ASPRS*. <https://www.asprs.org/divisions-committees/lidar-division/laser-las-file-format-exchange-activities> (Erişim Tarihi: 07 Mayıs 2025).
- URL-2: *Point Cloud Library 0.0 documentation*. https://pcl.readthedocs.io/projects/tutorials/en/latest/normal_estimation.html (Erişim Tarihi: 08 Mayıs 2025).
- URL-3: *Yenilikçi Mekansal Veri Üretim Yöntemleri — 2 | by Caner Guney | Medium*. <https://canerguney.medium.com/yenilik%C3%A7i-mekansal-veri-%C3%BCretim-y%C3%B6ntemleri-2-b35885215fe> (Erişim Tarihi: 07 Mayıs 2025).
- URL-4: Nielsen, M. *Neural Networks and Deep Learning*. <http://neuralnetworksanddeeplearning.com> (Erişim Tarihi: 07 Mayıs 2025).
- URL-5: xView – GitHub. <https://github.com/DIUX-xView> (Erişim Tarihi: 10 Nisan 2025).
- URL-6: *Princeton ModelNet*. <https://modelnet.cs.princeton.edu/> (Erişim Tarihi: 07 Mayıs 2025).

URL-7: *ShapeNet*. <https://shapenet.org/> (Erişim Tarihi: 08 Mayıs 2025).

URL-8: *Semantic3D*. <https://www.semantic3d.net/> (Erişim Tarihi: 07 Mayıs 2025).

URL-9: *VCC*. <https://vcc.tech/UrbanScene3D> (Erişim Tarihi: 07 Mayıs 2025).