

ANKARA ÜNİVERSİTESİ  
FEN BİLİMLERİ ENSTİTÜSÜ

DOKTORA TEZİ

DERİN ÖĞRENME YÖNTEMLERİNİ KULLANARAK SİSMİK HIZ  
MODELLERİNİN TERS ÇÖZÜMÜ

Mojtaba NAJAFIKHATOUNABAD

JEOFİZİK MÜHENDİSLİĞİ ANABİLİM DALI

ANKARA  
2025

Her hakkı saklıdır

# ÖZET

Doktora Tezi

## DERİN ÖĞRENME YÖNTEMLERİNİ KULLANARAK SİSMİK HIZ MODELLERİNİN TERS ÇÖZÜMÜ

Mojtaba NAJAFIKHATOUNABAD

Ankara Üniversitesi  
Fen Bilimleri Enstitüsü  
Jeofizik Mühendisliği Anabilim Dalı

Danışman: Prof. Dr. Selma KADIOĞLU  
Eş danışman: Doç. Dr. Hacer YALIM KELEŞ

Sismik arama yönteminde ölçülen verilerden yeraltı hız modelinin kestirimi doğrusal olmayan kötü durumlu ters çözüm problemidir. Bu problemin çözümü için farklı yöntemler kullanılmaktadır. Bu tez çalışmasında sismik hız modeli ters çözümüne yönelik derin öğrenme tabanlı bir yaklaşım sunulmaktadır.

Bu tez çalışmasında derin model olarak önerilen sismik hız modeli ters çözüm ağı (SVInvNet) yoğun bloklarla güçlendirilmiş çok bağlantılı bir kodlayıcı- kod çözücü yapısı içeren yeni bir mimari sunmaktadır. Uçtan uca sismik atış kesitlerini yer altı hız modellerine dönüştüren SVInvNet doğrusal olmayan bu dönüşümde kritik önemi olan sismik zaman serisi verilerini etkili bir şekilde işlemek üzere ayarlanmıştır.

Eğitim ve test veri kümesi olarak çok katmanlı, faylı ve tuz domu içeren çeşitli sismik hız modelleri ve her birinde hesaplanan sismik atış kesitleri hazırlanmıştır. Eğitim veri kümesi sayı olarak alt kümelere ayrılmış ve sismik verilere tutarlı ve rastgele gürültü türleri eklenmiştir. Böylece eğitim veri setinin sayısının ve farklı çevresel gürültü türlerinin eğitim süreci üzerindeki etkileri incelenmiştir.

SVInvNet'in performansı ayrıca OpenFWI veri kümesi ve Marmousi hız modelinden alınan hız modelleri üzerinde de denenmiştir. Elde edilen sonuçlar temel model olarak kullanılan InversionNet derin model ve tam dalga şekli ters çözüm yöntemi (FWI) ile karşılaştırılmıştır. Karşılaştırmalı analiz, bu tezin bir sonucu olarak önerilen mimarinin etkinliğini açıkça göstermektedir.

**Nisan 2025, 94 sayfa**

**Anahtar Kelimeler:** Derin Öğrenme, Sismik Ters-Çözüm, Tam Dalga Şekli Ters Çözüm, Sismik Hız Modeli, SVInvNet

# ABSTRACT

PhD Thesis

SEISMIC VELOCITY INVERSION UTILIZING DEEP LEARNING METHODS

Mojtaba NAJAFIKHATOUNABAD

Ankara University  
Graduate School of Natural and Applied Sciences  
Department of Geophysical Engineering

Supervisor: Prof. Dr. Selma KADIOĞLU  
Assoc. Supervisor: Assoc. Prof. Dr. Hacer YALIM KELEŞ

In seismic exploration, estimating the subsurface velocity model from acquisition data is a nonlinear and ill-posed inversion problem. Various methods have been used to solve this problem. In this thesis, a deep learning-based approach is presented for seismic velocity model inversion.

As the deep model proposed in this thesis, the Seismic Velocity Inversion Network (SVInvNet) introduces a novel architecture that incorporates a multi-connection encoder-decoder structure enhanced with dense blocks. SVInvNet is designed to effectively process seismic time-series data, which is crucial for this nonlinear transformation, and it converts seismic shot gathers into subsurface velocity models in an end-to-end manner.

For training and testing datasets, various seismic velocity models containing multi-layered, faulted, and salt dome structures are prepared, along with their corresponding seismic shot gathers. The training dataset was divided into subsets, and both coherent and random noise are added to the seismic data. This allowed for an investigation into the effects of the size of the training dataset and different types of ambient noise on the training process.

The performance of SVInvNet is further evaluated using the OpenFWI dataset and models derived from the Marmousi velocity model. The obtained results are compared with InversionNet, which is used as the baseline model, and the full-waveform inversion (FWI) method. The comparative analysis clearly demonstrates the effectiveness of the proposed architecture as an outcome of this thesis.

**April 2025, 94 pages**

**Key Words:** Deep Learning, Seismic Inversion, Full Waveform Inversion, Seismic Velocity Model, SVInvNet

## TEŞEKKÜR

Öncelikle, doktora sürecim boyunca beni yönlendiren, yardımlarını esirgemeyen ve değerli katkıları ile bu çalışmanın başarılı bir şekilde sonuçlanmasını sağlayan danışman hocalarım Sayın Prof. Dr. Selma KADIOĞLU ve Sayın Doç.Dr. Hacer YALIM KELEŞ'e şükranlarımı sunarım. Zaman ayırıp bilgi ve birikimleri ile akademik hayatıma tuttukları ışığın izi bilimsel çalışma kültürü olarak tüm hayatım boyunca devam edecektir.

Ayrıca, tez izleme komitesi üyeleri Sayın Prof. Dr. Bülent KAYPAK ve Sayın Doç. Dr. YILMAZ AR'a, kıymetli görüşleri ve yol gösterici önerileri için teşekkür ederim. Çalışmamın her aşamasında sundukları yapıcı eleştiriler ve destekleri, tezimin daha nitelikli hale gelmesine önemli katkı sağlamıştır.

Sadece tez çalışması sürecinde değil, bütün eğitim hayatım boyunca bana inanıp manevi ve maddi destekleri ile yanımda olan aileme en içten teşekkürlerimi sunarım. Özellikle sevgisi ve özverisi ile her zaman yanımda olan kıymetli anneme ayrıca teşekkür etmek isterim.

Doktora ve tez çalışmam boyunca bana moral verip bilgi ve deneyimlerini paylaşarak yardımcı olan değerli arkadaşlarım Cansu ARICAN, N. Yıldırım GÜNDOĞDU, ve Begüm KOCA'ya teşekkür ederim. Akademik ve kişisel anlamda samimi destekleri, bazen zorlu geçen bu süreci daha keyifli ve verimli hale getirmiştir.

Son olarak bu tez çalışmasının tamamlanmasında emeği geçen herkese en içten teşekkürlerimi sunarım.

Mojtaba NAJAFIKHATOUNABAD  
Ankara, Nisan 2025

## İÇİNDEKİLER

TEZ ONAY SAYFASI	
ETİK.....	i
ÖZET .....	ii
ABSTRACT .....	iii
TEŞEKKÜR.....	iv
SİMGELER DİZİNİ.....	vi
ŞEKİLLER DİZİNİ.....	vii
ÇİZELGELER DİZİNİ.....	x
1. GİRİŞ .....	1
1.1 Literatür Özeti .....	4
2. SİSMİK TERS ÇÖZÜM ve DERİN ÖĞRENME .....	11
2.1 Sismik Ters Çözüm .....	11
2.2 Derin Öğrenme .....	17
3. EĞİTİM VERİ KÜMESİ .....	22
3.1 Hız Modelleri.....	24
3.2 Sismik Verileri.....	29
3.3 Sismik Verilere Rastgele ve Tutarlı Gürültülerin Eklenmesi .....	33
3.4 OpenFWI Eğitim Veri Kümesi .....	35
3.5 Marmousi Tabanlı Alt Modeller Kümesi .....	37
4. DERİN MODELİN TASARIMI .....	39
4.1 Evrişimli sinir ağları (CNN'ler) .....	40
4.1.1 Oto-kodlayıcı ağ .....	43
4.1.2 Çok bağlantılı ağ .....	45
4.2 InversionNet Derin Modelin Mimarisi .....	47
4.3 SVInvNet Derin Modelin Tasarımı .....	49
4.3.1 Yoğun blokların yapısı.....	50
4.3.2 Kodlayıcı ağın özellikleri.....	51
4.3.3 Kod çözücü ağın özellikleri.....	52
5. EĞİTİM AŞAMASI .....	55
5.1 InversionNet ( $\alpha$ ), ( $\beta$ ) ve ( $\gamma$ ).....	56
5.2 SVInvNet.....	60
5.3 SVInvNet - OpenFWI Veri Kümesi .....	75
5.4 FWI Yöntemi ile Karşılaştırma .....	76
5.5 Marmousi Tabanlı Hız Modelleri .....	80
6. TARTIŞMA VE SONUÇLAR .....	82
KAYNAKLAR .....	87
EK1 Akustik Ortamda Sismik Dalga Yayılımı.....	90
ÖZGEÇMİŞ.....	94

## KISALTMALAR DİZİNİ

FWI	Full Waveform Inversion
CNN	Convolutional Neural Network
TD	Train Dataset
2B	İki Boyutlu
DL	Deep Learning
DLI	DL-based Seismic Inversion
FD	Fault Detection
FCNN	Fully CNN
IMB	Impedance Model Building
RMB	Reflectivity Model Building
USGS	United States Geological Survey.



## ŞEKİLLER DİZİNİ

Şekil 1.1 InversionNet derin modelin performans örneği. (a) gerçek hız modeli, (b) derin modelin kestirimi, (c) gerçek hız modelinde çizilen siyah çizgi boyunca iki modelin hız profilleri (Wu vd., 2018) . . . . .	8
Şekil 1.2 Li vd. (2020) tarafından sunulan SeisInvNet ve InversionNet derin modellerin karşılaştırılması. Soldan sağa sırayla gerçek hız modeli, SeisInvNet'in sonucu, InversionNet'in sonucu ve her üç modelin siyah çizgi boyunca hız profilleri . . . . .	8
Şekil 1.3 Liu vd.(2021), tarafından geliştirilen SeisInvNet ağıın sunduğu sonuçlardan bir örnek (a) gerçek hız modeli, (b) SeisInvNet modelin kestirimi, (c) InversionNet modelin kestirimi ve (d) siyah çizgi boyunca hız profili . . . . .	9
Şekil 2.1 Sismik düz ve ters çözümün şematik gösterisi. (a) Sismik düz çözümü ve (b) derin öğrenme yöntemi ile sismik hız modelin kestirimi. $\mathcal{L}$ Sismik dalga alanı yayılma operatörü . . . . .	16
Şekil 2.2 (a) Biyolojik sinir hücresi, (b) biyolojik siniri yansıtan matematik sinir modeli, (c) matematik sinir modelin katsayıları ile girdi verilerin çarpma toplama sonrası elde edilen sonucun aktivasyon fonksiyonu ile değerlendirilmesi, (d) farklı aktivasyon fonksiyonları (Şekiller Lippmann (1987), ve van der Baan ve Jutten (2000) 'den alınmıştır) . . . . .	19
Şekil 3.1 Denklem 3.1-3.3'te sunulan fonksiyonlar kullanılarak farklı ara yüzeyler ve hız modellerin oluşturulması . . . . .	27
Şekil 3.2 Beş grup hız modeline ait örnek modeller. Her bir gruptan tabakalı, faylı ve tuz domu içeren alt gruplardan birer örnek yar almaktadır . . . . .	30
Şekil 3.3 Sismik hız modelleri ve örnek atış kesitleri . . . . .	33
Şekil 3.4 Hesaplanan sismik atış kesitine tutarlı ve rastgele gürültülerin eklenmesi. (a) Tutarlı gürültü, (b) Rastgele gürültü, (c) Gürültülerin toplamı, (d) Gürültüsüz atış kesiti ve (h) Gürültülü atış kesiti . . . . .	35
Şekil 3.5 OpenFWI veri kümesine ait örnek hız modeller. İlk satırda CurveVel-B ikinci satırda ise CurveFault-A veri setlerine ait örnekler yer almaktadır . . . . .	36
Şekil 3.6 Marmousi hız modelinden alınan hız modelleri veri kümesine ait örnek modeller . . . . .	38
Şekil 4.1 $3 \times 3$ boyutlu yapay sinir çekirdiği ile girdi verinin arasındaki evrişim işleminin şematik görseli . . . . .	41
Şekil 4.2 Genişleme oranı farklı oalan $3 \times 3$ boyutlu CNN çekirdekleri ile imgenin arasındaki evrişim işleminin görseli . . . . .	43

Şekil 4.3 Kodlayıcı-kod Çözücü evrişimli sinir ağlarının şematik çizimi . . . . .	45
Şekil 4.4 Beş katmanlı çok bağlantılı derin model örneği (Huang vd., 2017) . .	46
Şekil 4.5 InversionNet derin modelin mimarisi (Wu vd., 2018) . . . . .	47
Şekil 4.6 Üç katmandan oluşmuş yoğun blok tasarımı. Katmanlardaki imge boyutu sabittir $m \times n$ . . . . .	50
Şekil 4.7 Sismik hız modeli ters çözümü için tasarlanan SVInvNet derin model mimarisi . . . . .	53
Şekil 5.1 InversionNet derin modelin ( $\alpha$ ), ( $\beta$ ) ve ( $\gamma$ ) versiyonlarına ait eğitim ve doğrulama yanlgı enerjileri. Gürültüsüz TD-V eğitim veri kümesi kullanılmıştır . . . . .	57
Şekil 5.2 InversionNet derin modelin üç farklı versiyonunun kestirdiği hız modelleri ve gerçek hız model. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen hız profili yer almaktadır. Modeller gürültüsüz TD-V eğitim veri kümesi ile eğitilmiştir . . . . .	58
Şekil 5.3 InversionNet derin modelin üç versiyonuna ait eğitim ve doğrulama yanlgı enerjileri. Gürültülü TD-V eğitim veri kümesi kullanılmıştır . . . . .	59
Şekil 5.4 Gerçek hız modeli ve InversionNet derin modelin üç farklı versiyonunun sonuçları. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen hız profili yer almaktadır. Modeller gürültülü TD-V eğitim veri kümesi ile eğitilmiştir . . . . .	60
Şekil 5.5 SVInvNet derin modelin L1 ve SSIM yanlgı enerjilerine ait 5 farklı eğitim ve doğrulama eğrisi. Eğriler TD I-V gürültüsüz veriler ile eğitilen modellere aittir . . . . .	62
Şekil 5.6 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirdiği hız modelleri. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tabakalı modellerden rastgele seçilmiştir . . . . .	64
Şekil 5.7 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in sonuçları. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece faylı modellerden rastgele seçilmiştir . . . . .	65
Şekil 5.8 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirimi. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tuz domu içeren modellerden rastgele seçilmiştir . . . . .	66

Şekil 5.9 Gerçek hız modeli, Model-I, Model-III, ve Model-V'in kestirimleri ve fark kesitleri. Eğitim için gürültüsüz veri kümesi kullanılmıştır . . . . .	67
Şekil 5.10 SVInvNet derin modelin L1 ve SSIM yanlıgı enejisi eğrileri. Modeller I-V, sırasıyla gürültülü TD I-V kullanılarak eğitilmiştir . . . . .	69
Şekil 5.11 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirdiđi hız modelleri. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tabakalı modellerden rastgele seçilmiştir . . . . .	71
Şekil 5.12 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in ürettiđi çıktılar. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece faylı modellerden rastgele seçilmiştir . . . . .	72
Şekil 5.13 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in sonuçları. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tuz domu içeren modellerden rastgele seçilmiştir . . . . .	73
Şekil 5.14 Gerçek hız modeli, Model-I, Model-III, ve Model-V'in kestirimleri ve fark kesitleri. Eğitim için gürültülü veri kümesi kullanılmıştır . . . . .	74
Şekil 5.15 Gerçek hız modelleri ve onlara ait InversionNet, VelocityGAN ve SVInvNet derin modellerin kestirimleri. İlk iki satır CurveVel-B ve son iki satır ise CurveFault-A benchmark'larından rastgele seçilmiştir. Hız profilleri, gerçek hız modellerinde çizilen ilgili gri çizgilere aittir . . . . .	76
Şekil 5.16 FWI, Model-I ve Model-V'in kestirdiđi hız modelleri gösterilmektedir. Gerçek model dikkate alınarak en iyi sonuç Model-V'e aittir. FWI yönteminde kullanılan başlangıç hız modeli de sunulmaktadır . . . . .	79
Şekil 5.17 Gerçek hız modeli, SVInvNet ve InversionNet- $\gamma$ derin modellerin kestirimleri. Hız profilleri 1-2-3 sırası ile gerçek modelde beyaz, gri ve sarı renkli kesik şekilde çizilen şeritlere aittir . . . . .	81

## ÇİZELGELER DİZİNİ

Çizelge 1.1 Sismik ters çözüm alanında 2020 yılına dek yayımlanan farklı yaklaşımlı araştırmalar (Adler vd., 2020) . . . . .	6
Çizelge 1.2 Araya-Polo vd. (2018) tarafından yayımlanan çalışmadaki hız modellerinin özellikleri . . . . .	7
Çizelge 1.3 Wu vd. (2018) tarafından yayımlanan hız modellerin özellikleri . . . . .	8
Çizelge 1.4 Li vd. (2020) tarafından yayımlanan çalışmadaki hız modellerin ve sismik kesitlerin özellikleri . . . . .	9
Çizelge 1.5 Liu vd. (2021) tarafından yayımlanan çalışmadaki hız modellerin ve sismik kesitlerin özellikleri . . . . .	10
Çizelge 5.1 InversionNet $\alpha$ , $\beta$ ve $\gamma$ versiyonlarının gürültüsüz ve gürültülü test verileri üzerindeki performansı. Gürültüsüz ve gürültülü olarak TD-V eğitim veri kümesi kullanılmıştır . . . . .	57
Çizelge 5.2 SInvNet derin modelin test veri seti üzerindeki performansı. Model-I den Model-V dek gürültüsüz eğitim veri kümeleri, TD I-V, kullanılarak elde edilmiştir. InversionNet( $\gamma$ )TD-V veri kümesi ile eğitilmiştir . . . . .	61
Çizelge 5.3 SInvNet derin modelin test aşamasındaki performansı. TD I-V gürültülü veri kümeleri kullanılarak sırası ile Model I-V eğitilmiştir. Son satırda InversionNet( $\gamma$ )-V'in test aşamasındaki sonuçları sunulmaktadır . . . . .	68
Çizelge 5.4 Test kümesindeki dört CNN Modelinin performansı. Modeller, CurveVel-B (CV-B) ve CurveFault-A (CF-A) veri setleri kullanılarak eğitilmiştir. . . . .	75

## 1. GİRİŞ

Araştırmalı sismik yansıma yönteminde yeraltının yapısal, petrofiziksel haritalanması için yerin sismik hız dağılımının belirlenmesi gerekmektedir. Yeraltı hız modeli göç işleminin ve görüntüleme tekniklerinin uygulanmasında ve derinlikle ilgili öngörülen kesitlerde önemli rol almaktadır. Ortamın hız modeline göre yer içinde ilerleyen, yansıyan veya kırılan sismik dalga alanının alıcılarda kaydedilen parametreleri (varış zamanı, geliş fazı ve genlik değeri) hız modeli ile ilgili bilgi içermektedir. Sismik ters çözüm yöntemlerinde bu bilgiler kullanılarak ortamın hız modeli kestirilmektedir. Ancak sismik dalga yayılma denkleminde, hiperbolik kısmi diferansiyel denklemi ile tanımlı, sismik dalga alanı ile hız modeli arasındaki ilişki doğrusallık özelliği taşımamaktadır. Diğer bir deyişle yer içini tanımlayan hız modelinin değiştirilmesi sismik verilerin parametrelerini herhangi bir orantı veya doğrusal ilişki ile değiştirmemektedir. Bu nedenle sismik verileri kullanılarak hız modelin kestirilmesi doğrusal olmayan bir ters çözüm problemidir. Bununla birlikte hız modeli parametrelerinde uygulanan küçük bir değişim sismik verilerde büyük farklara neden olabilmekteyken parametreleri farklı olan iki hız modelinde hesaplanan/ölçülen sismik verilerin parametreleri eşit olabilmektedir. Bu nedenle de hız modelinin kestirilmesi problemi kötü durumlu (ill-posed) bir ters çözüm problemi olarak nitelenmektedir.

Doğrusal olmayan kötü durumlu sismik hız modelinin ters çözümü için farklı yöntemler geliştirilmiştir. Türev tabanlı yöntemler, çözüm parametrelerine yakın olduğu varsayılan bir başlangıç hız modelinden hareketle yinelemeli adımlarla mevcut sismik verileri tahmin edebilen bir hız modeline doğru yaklaşmaktadır. Bu yöntemde parametrelerin değişim yönü ve büyüklüğü veri çakıştırma tekniği ile hesaplanmaktadır. Global yöntemlerde farklı algoritmalar kullanılarak ölçülen verilere en yakın değerleri üretebilen model parametrelerinin bulunması için hız modeli alanında arama yapılmaktadır.

Bilgisayar teknolojisinin ve yapay sinir ağları algoritmalarının geliştirilmesi ile bir-

likte giderek yapay zekanın alt bölümü olan derin öğrenme tekniği de çözüm yöntemlerinin arasında yer almaktadır. Aslında sismik ters çözüm alanında yapay sinir ağları kullanılarak yapılmış olan denemelerin geçmişi türev tabanlı yöntemlerin geliştirme zamanlarına kadar ulaşmaktadır (Roethe ve Tarantola, 1991). Ancak çağın hesaplama makinelerinin kısıtlı olması bu yöntemin geniş çapta yayılmasına mâni olmuştur. Derin öğrenme (deep learning-DL) yönteminde yapay sinir ağları kullanılarak sismik verileri hız modeline dönüştüren ağırlık veya katsayı veri kümesi hesaplanmaktadır.

Deneyimli (supervised) derin öğrenme yönteminde dönüştürme işlemi sağlayabilen katsayı veri kümesinin hesaplanması için yeterli sayıda girdi-çıkıtı (ground truth) şeklinde etiketlenmiş eğitim veri çifti hazırlanmalıdır. Çeşitli girdilerin ilgili çıktılara dönüştürülmesi ile yapay sinir ağlarındaki katsayıların hesaplanması eğitim süresini oluşturmaktadır. Derin modelin mimarisi olarak tanımlanan yapay sinirlerin türü, çıkarılan imgelerin boyutu, sinir ağı katman sayısı ve onların arasındaki bağlantıların düzeni çözülmesi hedeflenen probleme uygun bir şekilde tasarlanmalıdır. Kısacası deneyimli derin öğrenme yönteminde uygun eğitim veri kümesi ve derin model tasarımı problemin çözümü için önem taşımaktadır.

Bu tez çalışması kapsamında denetimli derin öğrenme yöntemleri kullanılarak sismik hız modeli ters çözümü yapılmakta ve yöntemi etkileyen önemli değişkenler araştırılmaktadır. Eğitim ve test veri kümesi olarak 18000 iki boyutlu( $2B$ ) hız modeli hazırlanmıştır. Farklı sismik stratigrafik örneklerin araştırılması sonucunda hazırlanan modeller tabaka sayısı açısından dörtten sekiz tabakaya dek değişerek beş gruba ve jeolojik yapı açısından tabakalı, faylı ve tuz domu içeren modellere ayrılarak da üç alt gruba bölünmektedir. Toplamda *on beş* alt gruptan oluşan hız modeli veri kümesinde her bir alt gruptan 1200 model üretilmiştir.

Her bir hız modeli için  $2B$  akustik varsayımı ile otuz dört alıcıdan oluşan yirmi sismik atış kesiti hesaplanarak girdi verileri hazırlanmıştır. Elde edilen atış kesitlerinde sayısal hesaplamadan kaynaklanan olası gürültü dışında herhangi bir istenmeyen veri

bulunmamaktadır. Ancak düzenli (coherent) ve rastgele (stochastic) gürültülerin derin model performansı üzerindeki etkisinin araştırılması amacıyla da hazırlanan sismik atış kesitlerine her iki gürültü türü eklenerek ayrıca bir gürültülü veri kümesi de elde edilmiştir. Böylece gürültülü ve gürültüsüz olmak üzere, her biri 18000 '*sismik atış kesitleri - hız modeli*' veri çiftinden oluşan iki farklı sentetik veri kümesi elde edilmiştir. Bu veri kümesi derin modelin eğitim ve test aşamasında kullanılmıştır.

Derin öğrenme yöntemi ile gerçekleştirilen sismik ters çözüm işleminde eğitim veri kümesinin sayısı olarak etkisinin araştırılması amacıyla, test veri kümesi sabit tutularak eğitim işlemi sayısı farklı olan beş eğitim veri seti ile gerçekleştirilmiştir. Bunun için hazırlanan sentetik verilerden 750, 1500, 3000, 4500 ve 6000 örnek rastgele bir şekilde ayıklanarak beş farklı eğitim veri kümesi ve bütün denemelerde sabit kalan 12000 örnekten oluşan test veri seti ayrılmıştır. Veri ayıklama veya denektaşları (benchmark) hazırlama işlemi hem gürültülü hem de gürültüsüz veri kümeleri üzerinde uygulanmıştır.

Girdi olarak sunulan sismik atış kesitlerini etiketli hız modeline dönüştürebilen (mapping) uygun derin modelin tasarlanması için farklı mimariler ve sinir ağları yapıları araştırılarak SVInvNet derin model tasarımı hazırlanmıştır. Her bir hız modeli kestiriminde derin modelin ayırt etmesi gereken parametreler tabaka sayısı, arayüzey şekli, tabakaların hız değeri, fay ve tuz domu yapıların şekil ve konumu olarak tanımlanabilmektedir. Adı geçen parametrelerin kestirimi derin modelin sismik atış kesitlerdeki üst ve alt seviyeli (high-level and low-level features) bilgilere ulaşması ile mümkündür. Bunun için de tasarlanan derin modelin katman sayısı veya derinlik boyutu yeterince büyük olmalıdır. Ancak giriş ve çıkış katmanlarının arasındaki bilgi akışı ister derinlik ister katmanlar arası geçiş şekli nedeni ile de kısıtlanmamalıdır. Etkin bir bilgi akışı tüm katmanlardaki katsayı veri kümesini amaçlanan çıktıyı en az hatayla üretecek yönde değiştirebilmektedir. Bu etkenler dikkate alınarak SVInvNet derin model tasarımında yoğun bloklara (dense block) bölünmüş kodlayıcı-kod çözücü (encoder-decoder) mimarisi kullanılmıştır.

Tasarlanan derin modelle elde edilen sonuçlar optimize edilmiş InversionNet (Wu vd. 2018) derin modelin ve tam dalga şekli ters çözüm (full waveform inversion - FWI) yöntemin sonuçları ile karşılaştırılmıştır. Ayrıca açık kaynak olarak yayınlanan OpenFWI (Deng vd. 2022) veri kümesi kullanılarak SVInvNet derin modelin eğitim ve test aşamasındaki performansı farklı sentetik veri kümesinde de denenmiştir. Söz konusu denemenin test aşamasındaki elde edilen sonuçları araştırmacıların yayınladığı üç farklı derin modelin sonuçları ile karşılaştırılmıştır (Najafi Khatounabad vd., 2020). Bu denemelerin yanısıra yeraltı hız dağılımını daha gerçekçi bir şekilde yansıtan Marmousi hız modelinden alınan hız modelleri de kellanılarak SVInvNet derin modelin performansı değerlendirilmiştir.

## 1.1 Literatür Özeti

Sismik ters çözüm alanında ilk defa Roethe ve Tarantola (1991) sinir ağlarını eğiterek bir boyutlu ( $1B$ ) hız modelini kestirebilmişlerdir. Bu çalışmada ‘sismik yansıma dalgaların varış zamanı - tek tabakalı hız modelin derinlik ve hız değeri’ şeklinde etiketlenerek eğitim veri çifti olarak kullanılmıştır. Eğitim aşamasında kullanılmayan girdiler için düşük hata payı ile uygun hız modellerin kestirilmesi araştırmacılar tarafından sinir ağlarının asıl başarısı olarak değerlendirilmiştir. Tam dalga şekli ters çözüm (FWI) yöntemin öncülerinden olan Tarantola çalışma arkadaşı ile yayınladığı bu makalede sinir ağlarını geleneksel ters çözüm yöntemlerinin bir alternatifi olarak tanıtmıştır.

1990’lı yıllarda dalga şekli tanımlama ve ilk varış zamanlarının derlenmesinde (Murat ve Rudman, 1992; McCormack vd., 1993), elektromanyetikte (Poulton vd., 1992), manyetotellürik yönteminde (Zhang ve Paulson, 1997) ve kuyu logu analizinde (Huang vd., 1996) olduğu gibi jeofiziğin farklı alanlarında sinir ağları yöntemi araştırmacılar tarafından kullanılmıştır. Ancak hesaplama olanaklarının kısıtlı olması ve sinir ağları algoritmalarının yeterince geliştirilmemesi nedeniyle bu yöntem uygulamalı çalışmalarda senelerce göz ardı edilmiştir. Zaman ilerleyince bir taraftan sinir ağları algoritmalarındaki gelişmeler diğer taraftan ise bilgisayar teknolojisindeki ilerleme-

ler bu yöntemin tekrar aktif arařtırmaların konusu haline getirmiřtir.

Günümüzde farklı disiplinlerde olduđu gibi, jeofizikte ve sismik ters çözüm alanında da bu yöntemin kullanımı arařtırmacılar ve yazılım geliřtirici firmalar tarafından ele alınmaktadır. Adler vd. (2020) tarafından yayınlanan derleme makalede sismik ters çözüm alanında 2020 yılına dek yapılmıř olan arařtırmaların önemli bir kısmı çizelge řeklinde yayımlanmıřtır. Bu tez çalışmasında daha detaylı olarak incelenmiř olan arařtırmalara geçmeden önce alanda yapılmıř olan bilimsel çalışmaların kronolojik süreci ile ilgili genel bir izlenimin oluřması amacıyla Adler vd. tarafından derlenen bilgiler Çizelge1.1’de sunulmaktadır.

Lewis vd. (2017) tuz domunun etkisi nedeni ile altındaki bölgelerin görüntüleme sorununun giderilmesi üzerine yaptıkları çalışmada derin öğrenme aracılıđıyla FWI yöntemi için iyi bir bařlangıç hız modeli oluřturmuřlardır. Böylece sismik kesitlerde bulanık bir bölge olarak ortaya çıkan tuz domunun alt kısmı daha yüksek bir çözünürlükle görüntülenmiřtir. Bu arařtırmada eğitim veri kümesi Meksika körfezi’nde yer alan hidrokarbon arama bölgesindeki tuz domu yapısını içeren gerçek veriler (sondajlar, kuyu logları ile denenmiř senelerce toplanan sismik verileri) kullanılarak yığma sonrası sismik kesitleri řeklinde hazırlanmıřtır. Bu arařtırmada derin model olarak evriřimli sinir ađları (convolutional neural network-CNN) mimarisi kullanılmıřtır.

Araya-Polo vd. (2018) GeoDNN adını verdikleri derin öğrenme ađını eğiterek iki boyutlu sentetik hız modellerini kestirebilmiřlerdir. Çalışmalarında  $2B$  sentetik hız modellerinde akustik sismik dalga denklemini çözerek yapay sismik kesitleri elde etmiřlerdir. Elde edilen kesitlerde ortak orta nokta (CMP) hız analizi yapılarak her bir hız modeli için CMP hız küpü hazırlanmıřtır. Böylece derin öğrenme ađının eğitimi için girdi olarak CMP hız küpü ve hedef (ground truth) olarak da hız modeli kullanılmıřtır.

Bu arařtırmada eğitilen ađ tabakalı hız modellerinin tabaka sayısını dođru kesti-

Çizelge 1.1 Sismik ters çözüm alanında 2020 yılına dek yayınlanan farklı yaklaşımlı araştırmalar (Adler vd., 2020)

Kaynak	Yık	Uygulama	Derin Model Mimarisi
Lewis vd.	2017	Tuz domu/DL-FWI	CNN
Richardson	2018	VMB/DL-FWI	RNN
Araya-Polo vd.	2018	VMB/DLI	cNN
Kim vd.	2018	RMB/DLI	MLP
Wang vd.	2018	VMB/DLI	CNN kodlayıcı-kod çözücü
Wu vd.	2018	VMB ve FD/DLI	CNN kodlayıcı-kod çözücü
Alfarraj vd.	2018	IMB/DLI	GRU tabanlı CNN
Das vd.	2018	IMB/DLI	CNN
Biswas vd.	2019	RMB/DLI	Fiziğe dayalı CNN
Yang vd.	2019	VMB/DLI	U-Net kodlayıcı-kod çözücü
Adler vd.	2019	VMB/DLI	RNN
Mao vd.	2019	VMB/DLI	CNN
Zheng vd.	2019	VMB ve FD/DLI	CNN
Araya-Polo vd.	2019	VMB/DLI	GAN
Das vd.	2019	IMB/DLI	CNN
Duque vd.	2019	VMB/DLI	Koşullandırılmış GNN
Alfarraj vd.	2019	IMB/DLI	Fiziğe dayalı RNN
Ovcharenko vd.	2019	LFE/DL-FWI	FCNN
Wang vd.	2019	IMB/DLI	Cycle GAN
Fabien-Ouellet vd.	2020	VMB/DLI	RNN kodlayıcı-kod çözücü
Park ve Sacchi	2020	VMB/DLI	FCNN
Li vd.	2020	VMB/DLI	CNN kodlayıcı-kod çözücü
Mosser vd.	2020	VMB/DL-FWI	Wasserstein GAN
Sun ve Deman vd.	2020	LFE/DL-FWI	CNN
Wu vd.	2020	IMB/DLI	Residual FCNN
Wang vd.	2020	VMB/DLI	CNN kodlayıcı-kod çözücü
Zhang ve Lin	2020	VMB/DLI	Wasserstein GAN

DL-FWI: DL ile FWI'nın performansının iyileştirilmesi; DLI: Tamamen DL'ye dayalı sismik Evirtim; FCNN: Tamamen CNN; FD: Fay bulma; IMB: Empedans modeli oluşturma; RMB: Yansıma modeli oluşturma

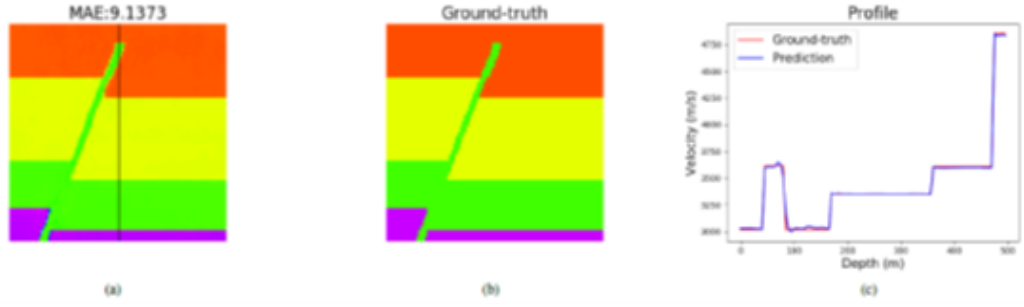
Çizelge 1.2 Araya-Polo vd. (2018) tarafından yayınlanan çalışmadaki hız modellerinin özellikleri

Boyut	Tabaka Sayısı	Hız Değerleri	Jeolojik Yapı
$35 \times 45$	3 – 8	2 – $4km/s$	Fay ve bazı modellerde tuz domu

rebilmiştir. Ancak hız farkı yüksek olan ara yüzeyleri daha yumuşak bir geçiş ile oluşturabilmiştir. Fay ve tuz domu içeren hız modellerinde faylar belirsiz ve tuz domları daha düşük hızla kestirilmiştir. Araştırmacılar bilgisayar imkanlarının ve eğitim verilerinin yetersizliğini istedikleri sonuca erişememelerinin nedeni olarak ifade etmişlerdir. Hazırlanan hız modellerinin özellikleri Çizelge 1.2’de sunulmaktadır.

Sonraki araştırmacılar üzerinde herhangi bir sismik veri işlem yapılmamış sismik atış kesitlerini girdi şeklinde kullanarak derin ağı eğitim sürecini gerçekleştirmişlerdir. Diğer bir deyişle, eğitim veri kümesi ‘sismik atış kesitleri-hız modeli’ şeklinde hazırlanmıştır. Bu sayede veri işleme zamanı, hesaplama ve kullanıcı hata payı olduğunca azaltılmıştır. Yang ve Ma (2019) UNet tasarımını kullanarak tuz domu içeren yatay tabakalardan oluşan hız modellerin öğrenilmesi için sismik atış kesitlerini girdi olarak ağı sunmuşlardır. Elde edilen görece başarılı sonuçlar daha karmaşık hız modellerin sismik atış kesitleri ile eğitilebilirliği konusunda önem kazanmaktadır.

Wu vd. (2018) InversionNet adını verdikleri derin öğrenme ağına sismik atış kesitlerini girdi olarak sunmuş ve fay içeren hız modellerini eğitmişlerdir. Kodlayıcı-kod çözücü mimarisi kullanılarak tasarlanan InversionNet, dört tabakalı yatay ara yüzeyli fay içeren hız modellerini belirli başarıyla kestirebilmiştir (Şekil 1.1). Bu çalışmada tasarlanan derin öğrenme ağının kodlayıcı biriminde iki boyutlu CNN çekirdekleri kullanılarak sismik atış kesitleri sayı olarak artan ve boyut olarak da küçülen imgelere dönüştürülmektedir. Kod çözücü biriminde ise giderek birleştirilen imgelerin boyutu ters evrimsel (dekonvolüsyonel) çekirdeklerin aracılığı ile büyütülüp çıkıştaki hız modeli oluşturulmaktadır. Bu çalışmada yeraltı gerçek modellerin yapısından farklı olarak hız modellerin ara yüzeyleri yatay ve düz şeklinde tasarlanmıştır. Hız modellerinin ve sismik atış kesitlerin özellikleri Çizelge 1.3’te sunulmaktadır.

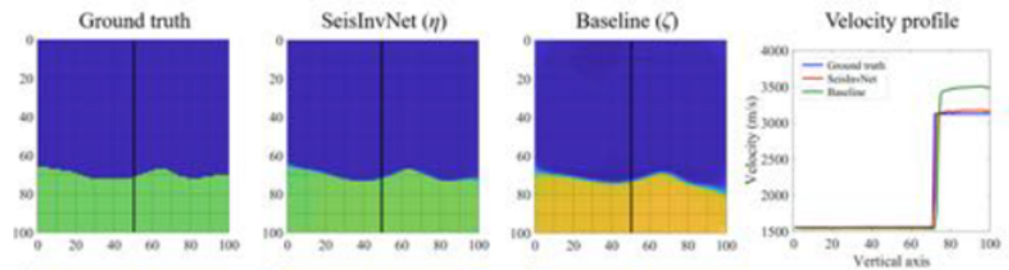


Şekil 1.1 InversionNet derin modelin performans örneği. (a) gerçek hız modeli, (b) derin modelin kestirimi, (c) gerçek hız modelinde çizilen siyah çizgi boyunca iki modelin hız profilleri (Wu vd., 2018)

Çizelge 1.3 Wu vd. (2018) tarafından yayınlanan hız modellerin özellikleri

Boyut	Tabaka Sayısı	Hız Aralığı	Kaynak ve Alıcı Sayısı	Model Sayısı
$100 \times 100$	2 – 4	3 – 5 km/s	3 kaynak, 32 alıcı	60000

Li vd. (2020) InversionNet derin modeli baz alarak SeisInvNet adı ile yeni bir derin model tasarımı sunmuşlardır. SeisInvNet eğimli-dalgalı ara yüzeyleri olan hız modellerini kestirebilmiştir (Şekil 1.2). Ancak modellerinde fay ve tuz domu yapıları bulunmamaktadır. SeisInvNet derin modelin ilk katmanlarında tam bağlı (fully connected-FC) katmanlar kullanılarak her bir sismik ize; komşu izleri, veri toplama dizilimi ve yer aldığı profilin genel durumu ile ilgili bilgiler eklenmiştir.



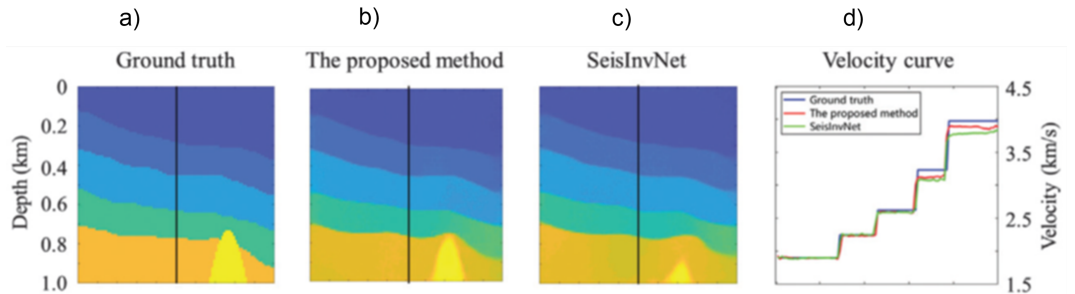
Şekil 1.2 Li vd. (2020) tarafından sunulan SeisInvNet ve InversionNet derin modellerin karşılaştırılması. Soldan sağa sırayla gerçek hız modeli, SeisInvNet’in sonucu, InversionNet’in sonucu ve her üç modelin siyah çizgi boyunca hız profilleri

Çizelge 1.4 Li vd. (2020) tarafından yayınlanan çalışmadaki hız modellerin ve sismik kesitlerin özellikleri

Boyut	Tabaka Sayısı	Hız Aralığı	Kaynak ve Alıcı Sayısı	Model Sayısı
$100 \times 100$	2 – 5	1,5 – 4km/s	20 kaynak, 100 alıcı	12000

SeisInvNet ile elde edilen sonuçlar temel model (baseline) olarak adlandırılan InversionNet'in sunduğu sonuçlar ile karşılaştırılmıştır. Şekil 1.3'te gerçek hız modeli, SeisInvNet ve InversionNet ile elde edilen hız modeli gösterilmektedir. Bu Şekil'de her bir modele ait siyah çizgiyle belirtilmiş hız profili de karşılaştırılmıştır. Hız profili karşılaştırmasında mavi çizgi gerçek, kırmızı SeisInvNet ve yeşil olan InversionNet ile elde edilmiş hız modellerine aittir. Burada kırmızı çizgi nerdeyse mavi çizgi ile örtüşmekteyken yeşil olan ikinci tabakanın hız değerini kestirememiştir. Böylece Li vd. (2020) SeisInvNet derin modelin InversionNet derin modele göre daha iyi sonuçların sunduğunu göstermeye çalışmışlardır. Araştırmacılar tarafından kullanılan eğitim veri kümesinin, hız modelleri ve sismik kesitleri, özellikleri Çizelge 1.4'te sunulmaktadır.

Liu vd. (2021) fay ve tuz domu gibi jeolojik yapıları içeren, ara yüzeyleri düz olmayan çok tabakalı hız modellerini derin öğrenme ağına eğitebilmek için SeisInvNet'in giriş bölümünü geliştirmişlerdir. Şekil 1.3' te Liu vd. (2021) tuz domu içeren hız modeli için yayınladıkları sonuçlar yer almaktadır. Bu çalışmada sismik verileri alıcı kesitleri şeklinde de düzenlenmiş ve sismik atış kesitleri ile birlikte girdi olarak



Şekil 1.3 Liu vd.(2021), tarafından geliştirilen SeisInvNet ağıın sunduğu sonuçlardan bir örnek (a) gerçek hız modeli, (b) SeisInvNet modelin kestirimi, (c) InversionNet modelin kestirimi ve (d) siyah çizgi boyunca hız profili

Çizelge 1.5 Liu vd. (2021) tarafından yayınlanan çalışmadaki hız modellerin ve sismik kesitlerin özellikleri

Boyut	Tabaka Sayısı	Hız Aralığı	Kaynak ve Alıcı Sayısı	Model Sayısı
$100 \times 100$	5 – 9	1,5 – 4km/s	20 kaynak, 32 alıcı	18000

ağa sunulmuştur.

Liu vd. (2021) hız modellerinin ara yüzeylerini biri doğrusal ikisi sinüzoidal üç fonksiyonun toplamından oluşan bir denklemin parametrelerini değiştirerek oluşturmuşlardır. Hazırladıkları hız modellerin ve sismik kesitlerin özellikleri Çizelge 1.5'te sunulmaktadır.

Anlatılan çalışmalarda sismik hız modelin derin öğrenme ile kestirilmesi bir regresyon problemi gibi ele alınmaktadır. Ancak Simon vd. (2024) tarafından yapılmış olan karşılaştırmalı araştırmada zaman ortamında ( $V_{rms}$ ) hız değerlerinin kestirimi için regresyon ve sınıflandırıcı yöntemleri kullanılmıştır. Bu araştırmada yayınlanan sonuçlar sınıflandırıcı yöntemi ile kestirilen hız modellerin hata payının regresyon yöntemine göre daha az olduğunu göstermektedir. Araştırmalar daha güvenilir sonuçlara yönelik devam etmektedir.

Literatürde yer alan derin öğrenme yöntemleri ile sismik hız modellerin kestirimi için yapılmış olan çalışmaların ortak amacı, derin modellerle uçtan uca (sismik kesitlerden doğrudan hız modeline giden) daha gerçekçi hız modellerin kestirilmesidir. Hız modellerin tabaka sayısı, ara yüzey şekli, karmaşık jeolojik yapıları (fay ve tuz domu gibi) basit örneklerden başlayıp karmaşık ve gerçekçi modellere doğru evrilmektedir. Bununla birlikte uçtan uca hız modellerini kestirebilen derin modelin tasarımı yapay sinir ağlarındaki güncel gelişmelerden faydalanarak daha iyi performans ile sorunun çözülmesine odaklanmış durumdadır.

## 2. SİSMİK TERS ÇÖZÜM VE DERİN ÖĞRENME

Ölçülen verilerin özelliklerini etkilemiş olan bir değişkenin/değişkenlerin kestirilmesi ters çözüm tanımının kapsamında yer almaktadır. Bunun tersi yani bahsedilen değişkenin etkisini modelleyebilen matematik denklemi kullanılarak veriyi hesaplamak düz çözüm olarak adlandırılmaktadır. Jeofizik yöntemlerinde ölçülen veriler kullanılarak onları etkileyen değişkenlerin veri toplama alanındaki dağılımı farklı ters çözüm yöntemleri ile hesaplanmaktadır. Değişkenler ile ölçülen verilerin arasında fizik kurallarına dayalı matematik denklem kullanılarak düz çözüm veya modelleme yapılmaktadır (Schuster 2017, Wang 2016, Maurya vd., 2016).

Arama sismiği yönteminde sismik atış kesitlerine ait kaydedilmiş izlerin (dalga alanlarının) varış zamanı, genliği ve varış fazı ölçü alanının sismik hız dağılımından etkilenmektedir. Sismik veriler kullanılarak ortamın hız haritasının kestirilmesi sismik hız modeli ters çözümü olarak tanımlanmaktadır. Ters çözümün sonucunda elde edilen hız dağılımı kullanılarak hesaplanan sentetik verilerin ölçülen verilere oldukça yakın olması hedeflenmektedir. Gerçek ve kuramsal verilerin arasındaki hata payı ne kadar düşük olursa hesaplanan hız modeli bir o kadar başarılı sayılmaktadır (Sen ve Stoffa, 2013).

Bu bölümde sismik hız modelin akustik ortamda 2B terss çözümü ele alınarak problemin çözümü için kullanılan farklı yöntemler anlatılmaktadır. Sonraki bölümlere gerekli zeminin hazırlanması için yöntemlerin içinden derin öğrenme yöntemi daha detaylı bir şekilde ele alınmıştır.

### 2.1 Sismik Ters Çözüm

Sismik arama yönteminde yüzeyden yüzeye veri toplama tekniğinde yeryüzünde konumlanan kontrollü sismik kaynağı kullanılarak sismik dalga alanı yeraltına gönderilmektedir. Sismik dalga alanı hız değerleri farklı olan ara yüzeylerden yansıma,

saçılma ve kırılma nedeni ile yeryüzüne geri dönerek veri toplama dizilimine göre yüzeyde konumlanan alıcılarda zaman serisi olarak kaydedilmektedir. Alıcılarda kaydedilen veriler *sismik iz* veya sismik dalga alanı ve alıcı sırasına göre yan yana dizilmiş olan sismik izler ise *sismik atış kesiti* olarak tanımlanmaktadır. Sismik atış kesitlerinde her bir alıcıda kaydedilen sismik izde doğrudan gelen, yansıma, saçılma, kritik kırılma dalga fazları ve Rayleigh, tekrarlı yansımalar vb.  $x - z$  düşey düzlem üzerinde ilerleyen tüm dalga fazları ve diğer düşey düzlem üzerinde ilerleyen gürültüler yer almaktadır. Yeraltı modelini temsil eden sismik hız dağılım kesiti farklı dalga fazlarını yayılma özelliklerine göre farklı şekilde etkilemekte, varış zamanlarını, genliklerini ve eğriselliklerini etkilemektedir.

Hız dağılımı belli olan bir ortamda sismik dalga alanının yayılması ve farklı noktalardaki özellikleri düz çözüm ile hesaplanmaktadır. Akustik varsayımı ile 2B bir ortamda sismik dalga alanı yayılma denklemi ve akustik ortamın özellikleri Ek1'de sunulmaktadır. Matematik ifadeyi kolaylaştırmak amacıyla düz çözüm Şekil 2.1.a'da şematik olarak da gösterildiği gibi bir operatör,  $\mathcal{L}$ , aracılığı ile simgelenmektedir. Böylece sismik verilerle hız modeli parametreleri arasındaki ilişkiyi gösteren sismik dalga alanı hesaplama (düz çözüm) denklemi izleyen şekilde yazılabilir;

$$D = \mathcal{L}(m_v) \quad (2.1)$$

Burada,  $D$  sismik verinin ve  $m_v$  hız modelin parametrelerini göstermektedir. Diğer bir deyişle hız parametreleri bilinen bir ortamda  $\mathcal{L}$  operatörü o ortamdaki dalga yayılımı ve sismik verilerin parametrelerini hesaplayabilmektedir. Matematik bakış açısından Şekil 2.1.a'da  $\mathcal{L}$  operatörün yönünün değiştirilmesi ( $\mathcal{L}^{-1}$ ) sismik verilerden hız parametrelerinin,  $m_v$ , elde edilmesi anlamına gelmektedir. Ters çözüm olarak tanımlanan bu ifade Denklem 2.2 ile temsil edilmektedir.

$$m_v = \mathcal{L}^{-1}(D) \quad (2.2)$$

$\mathcal{L}$  operatörünün çözümü için sonlu farklar gibi çeşitli sayısal yöntemler kullanılabilir, ama  $\mathcal{L}^{-1}$  hesaplamak için herhangi bir doğrusal süreç veya denklem bulunmamakta-

dır. Ölçülen veya sentetik modellemede hesaplanan sismik verileri üretebilen uygun hız modelin kestirilmesi için  $\mathcal{L}^{-1}$  operatörünün çözülmesi gerekmektedir.

Dalga alanı yayılma denklemi hiperbolik denklemler sırasında yer almakta ve hız parametrelerinin değişimi ile sismik verilerin değişimi arasında herhangi bir orantı veya doğrusal ilişki bulunmamaktadır. Bu nedenle de  $\mathcal{L}^{-1}$  çözüm süreci doğrusal olmayan karmaşık bir ters çözüm problemidir. Bununla birlikte parametreleri farklı olan iki hız modeli aynı sismik verileri üretebilmekte veya hız modeli parametrelerindeki küçük bir değişim çok daha farklı sismik verilerin ortaya çıkmasına neden olabilmektedir. Bu ise sismik hız modellerin ters çözümünü kötü durumlu probleme dönüştürmektedir.

$\mathcal{L}^{-1}$  operatörün çözümü veya sismik hız modeli ters çözümü için farklı yöntemler geliştirilmiştir. **Türev tabanlı** ters çözüm yöntemleri bir başlangıç hız modelinin,  $\bar{m}_v$ , parametrelerini değiştirerek ölçülen sismik verileri öngörebilen hız modelini hesaplamaktadır. Bu yöntemde başlangıç hız modelinde hesaplanan kuramsal sismik verileri,  $\bar{D}$ , ölçülen sismik veriler,  $D$ , ile karşılaştırılarak aralarındaki farkın,  $E(D - \bar{D})$ , azalması hedeflenmektedir. Bunun için yanılgi enerjisinin,  $E$ , yerel türevleri sıfıra eşitlenerek yinelemeli adımlarla  $\bar{m}_v$  modelin parametrelerinin değişim yönü ve büyüklüğü hesaplanmaktadır. Yöntemin başarı payını büyük ölçüde etkileyen başlangıç modelin elde edilmesi farklı veri işlem aşamalarını gerektirmektedir. Kötü bir başlangıç hız modeli çözüm sürecini yerel minimumlara doğru yönelterek gerçek modelin elde edilmesine sebep olmaktadır.

Türev tabanlı yöntemler veri karşılaştırma süresinde sismik verilerin farklı değişkenlerini (varış zamanı veya tam dalga şekli) kullanmakta ve buna göre de sınıflanmaktadır. Seyahat zamanı tomografi yönteminde karşılaştırılan veriler varış zamanlarından ibarettir. Diğer bir deyişle bu yöntemde, yanılgi enerjisi alıcılarda kaydedilen verilerin varış zamanları ile onlara karşılık gelen kuramsal verilerin varış zamanları karşılaştırılarak hesaplanmaktadır (Iversen ve Gjoystdal, 1996; Stefani,1995). Yanılgi enerjisini azaltmaya çalışan tomografi yönteminde varış zamanları ışın yolu yönte-

miyle hesaplanmaktadır. Bu yöntemde kuramsal varış zamanları sismik dalganın kaynaktan alıcıya kadar ilerlediği varsayılan yol boyunca yavaşlık ( $1/\text{hız}$ ) değişkenine göre integrali alınarak hesaplanır (Schuster, 2017).

Düşük bileşenleri içeren geniş frekans bantlı sismik dalga alanının yeraltındaki davranışını, yüksek frekans varsayımına dayanan ışın yolu yaklaşımı ile modellemek gerçek hız modelinden uzaklaştıracak kadar büyük hatalara neden olabilmektedir (Williamson, 1991). Seyahat zamanı tomografi yöntemini geliştiren araştırmacılar hata payını küçültmek amacıyla ışın yolları hesabına saçılma dalgalarını da eklemişlerdir (Devaney, 1984). Elde edilen olumlu ilerlemelere rağmen sonucun gerçek hız modeline daha da yakınsaması sadece modelleme hatalarının giderilmesi ile olası değildir. Ters-çözüm aşamasında çakıştırmak için seçilen özellik/ler de önem taşımaktadır.

Ölçülen verilerin sadece varış zamanlarını kestirerek hesaplanan hız modeli diğer bir deyişle varış zamanlarını çakıştırmak için ilerleyen ters çözüm algoritması, en iyi ihtimalle sismik dalga alanının taşıdığı geometrik bilgilerden faydalanabilmekte ancak dinamik bilgileri kaçırmaktadır. Örneğin, bir tortul kayada tuzlu su yerine gaz da olsa yansıyan dalganın ilerleme hızı veya ona bağlı olarak seyahat zamanı önemli ölçüde değişim göstermeyebilir, ancak yansıma genlikleri parlak noktaların (bright spot) oluşabileceği büyüklüğünde bir farkla ortaya çıkabilmektedir (Wang ve Nur, 1990). Uygulama açısından da verilerin varış zamanını belirlemek zaman alıcı ve hata payı yüksek olan bir çalışma süresidir.

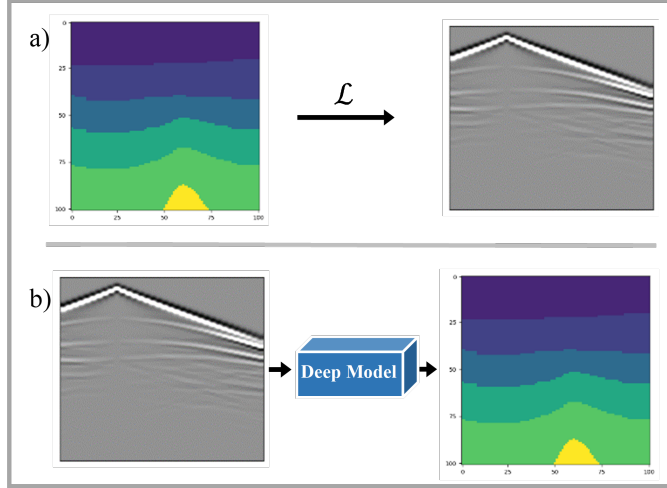
Sismik verilerin farklı değişkenleri hız modelin parametreleri ile ilgili farklı bilgileri içermektedir. Bu nedenle de araştırmacılar sadece varış zamanı değil, ölçülen sismik dalga alanının tamamını öngörebilen bir hız modeli ile sonuçlanan ters-çözüm yöntemi de geliştirmişlerdir. Bu yöntemde, verilerin varış zamanı, dalga genliği ve varış fazı diğer bir deyişle sismik dalganın tam formu kullanıldığı için tam dalga şekli ters çözüm olarak (full waveform inversion - FWI) adlandırılmaktadır.

Bu terim ilk kez tabakalı bir ortamın jeolojik yapısını belirlemek için sismograflardaki dalga alanlarının tamamını kullanan Pan vd. (1988) tarafından tanımlanmıştır. Daha önceki araştırmacılar, örneğin; Tarantola 1984, ve Gauthier vd. 1986 aynı işlemi dalga şekli ters çözüm (waveform inversion) olarak adlandırmışlardır. FWI yönteminde, varış zamanı değerleri (geometrik bilgiler) ile birlikte genlik değişiminin de (dinamik bilgiler) karşılaştırılması yapılmaktadır (Virieux ve Operto, 2009). Hız modelindeki bazı parametrelerin değişimi dalga şeklini, seyahat zamanı değerinden daha çok etkilemektedir (Schuster, 2017).

Bu nedenle FWI yöntemi karmaşık hız modellerini kestirmekte, yansıma sınırlarını konumlandırmada ve yüksek çözünürlük sağlamada başarılı sonuçlar sunmuştur (Warner vd. 2013, Ben-Hadj-Ali vd. 2008, Sirgue vd. 2008). Ancak sismik verilerin tam şeklinin karşılaştırılması ister en küçük kareler (least squares error) yöntemiyle olsun ister ortalama mutlak hata (mean absolute error) yöntemi ile sinüzoidal bir yanılğı enerjisi fonksiyonuna neden olmaktadır. Bu ise genel minimum noktasının yerel minimumlarla çevrelenmesi anlamına gelmekte ve başlangıç modelin önemini göstermektedir.

Düz çözümü temsil eden (2.1) denkleminde hız modeli değiştirilerek farklı sismik veriler elde edilebilir. Ölçülen sismik verileri üretebilen hız modelini farklı algoritmalar aracılığı ile bulmaya çalışan yöntemler **tümel** olarak tanımlanmaktadır. Kullanılan algoritmalar optimum  $m_v$  modelini bulmak için veri ortamında değil model ortamında arama yapmaktadır. Bu nedenle de çok büyük bellekler, özellikle de büyük modellerin çözüm sürecinde, gerekmektedir.

Bilgisayar teknolojisinin ve sinir ağları algoritmalarının gelişmesi ile birlikte son yıllarda derin öğrenme yöntemi de sismik hız modeli ters çözümü amacıyla kullanılmaktadır. Derin öğrenme yönteminde sinir ağları aracılığı ile sismik verilerini yeraltı hız modeli ile eşleştiren ağırlık veya katsayı veri kümesi hesaplanmaktadır. Diğer bir deyişle önceki yöntemlerde ters çözüm işlemi hız modelinde hesaplanan sismik verilerin değerlendirilmesi ile ilerlemekteyken derin öğrenme yönteminde  $\mathcal{L}^{-1}$  operatörün



Şekil 2.1 Sismik düz ve ters çözümün şematik gösterisi. (a) Sismik düz çözümü ve (b) derin öğrenme yöntemi ile sismik hız modelin kestirimi.  $\mathcal{L}$  Sismik dalga alanı yayılma operatörü

öğrenilmesi ile sismik verilerinden hız modelinin oluşturulması hedeflenmektedir. Bu işlemi gerçekleştirebilen katsayı veri kümesi eğitim olarak adlandırılan süreçte elde edilmektedir. Şekil 2.1.b'de şematik olarak gösterildiği gibi derin öğrenme sürecinin matematik ifadesi Denklem 2.3'te sunulmaktadır.

$$\mathcal{F}(D^n, \mathcal{W}) \rightarrow m_v^n \quad (2.3)$$

Burada sismik verilerini,  $D^n$ , hız modeli,  $m_v^n$ , ile eşleştiren derin öğrenme fonksiyonu  $\mathcal{F}$  olarak simgelenmektedir.  $\mathcal{W}$  ise yapay sinir ağlarındaki ağırlık veri kümesidir. Eğitim veri kümesinin toplam sayısı  $N$  olarak kabul edilirse  $n \in N$ ,  $\mathcal{F}$  fonksiyonu derinliği (katman sayısı) ve bağlantı haritası farklı olan sinir ağları katmanlarından oluşmaktadır.

Denetimli derin öğrenme yönteminde eğitim süreci önceden hazırlanmış yeterli sayıda ve kapsamlı nitelikte girdi ve amaç (ground truth) veri çiftleri ile gerçekleşmektedir. Eğitim veri kümesi olarak tanımlanan etiketlenmiş bu veri çiftleri bu tez kapsamında 'sismik veri-hız modeli' şeklinde sırayla girdi ve amaç verisi olmak üzere hazırlanmıştır. Sismik verileri hız modellerinin üst yüzeyindeki profillere ait atış kesitleri (shot gather) temsil etmektedir.

Derin modelin mimarisi ve sinir ağların türü çözümleni hedeflenen problemin özelliklerine uygun şekilde tasarlanmalıdır. Girdi olarak derin modele sunulan sismik verisi ile sinir ağlarında bulunan katsayıların arasında çarpma toplama işlemi ilk katmandan son katmana dek devam etmektedir. Derin modelin sonucu olarak son katmandan çıkan veri amaç verisi ile karşılaştırılarak aradaki fark yanlış enerji fonksiyonu ile hesaplanmaktadır. Söz konusu fark değerinin azaltılması için yanlış enerjisinin sıfıra eşitlenen türevi hesaplanarak geri yayılma yöntemi ile sinir ağlarına yayılarak katsayılar derin modelin sonucunu amaç veriye yaklaştıracak yönde değiştirilmektedir. Eğitim süreci bittikten sonra eğitilmiş derin modelin eğitim sürecinde girdi olarak kullanılmayan sismik verilere ait hız modelin düşük hata payı ile kestirmesi beklenilmektedir.

Özet olarak derin modelin mimarisi ve eğitim veri kümesinin sayısı ve kapsamı derin öğrenme yönteminin başarı oranını etkileyen iki önemli değişkendir. Bu tez kapsamında sismik ters çözüm işleminin derin öğrenme yöntemleri ile gerçekleştirilmesi için bir taraftan derin model mimarisi incelenerek özgün bir derin model mimarisi tasarlanmış diğer taraftan ise eğitim ve deneme (test) süreçleri için de yeterli sayıda ve kapsamlı biçimde sismik verileri-hız modeli şeklinde etiketlenmiş veri çiftleri hazırlanmıştır.

## 2.2 Derin Öğrenme

Tezin ilerleyen bölümlerinde kullanılan terimlere ve kavramlara uygun zeminin hazırlanması amacıyla bu bölümde derin öğrenme ile ilgili gerekli bilgiler özet olarak sunulmaktadır. Açıklanan konularla ilgili daha detaylı bilgilere erişmek isteyen okuyucular için ilgili kaynaklara yer verilmiştir.

Canlıların sinir ağı sisteminden esinlenerek geliştirilen ilk matematik (yapay) sinir modeli yaklaşık seksen yıl önce McCulloch ve Pitts (1943) tarafından tasarlanmıştır. Biyolojik sinir modeli, Şekil 2.2.a'da görüldüğü gibi, üç bölümden oluşmaktadır. Söz konusu bölümler, dış etkenlerin veya diğer sinirlerin ilettiği sinyalleri girdi olarak

algılayan **dendrit**, sinyalleri toplayıp değerlendiren **soma** ve aktarma koşullarını sağlayan sinyali diğer hücrelere ileten kısım yani **akson** olarak tanımlanmaktadır.

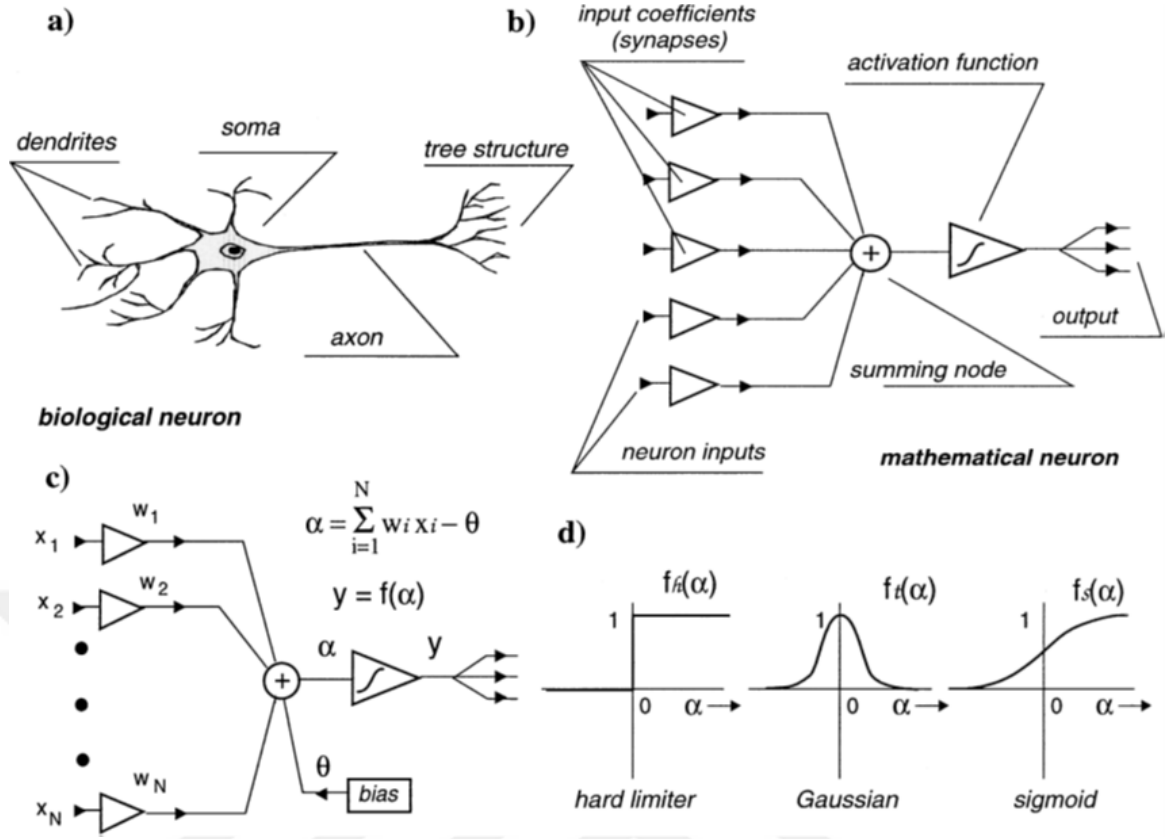
Yan yana dizilmiş olan binlerce sinir hücreleri A noktasında üretilmiş bir sinyalin B noktasına ulaşmasını sağlayarak sinir ağını oluşturmaktadır. Şekil 2.2.b'de biyolojik sinir yapısına benzer şekilde tasarlanmış olan matematik bir sinir modeli sunulmaktadır. Her bir yapay sinir modeli; giriş, afin dönüşüm (affine transformation) (Şekil 2.2.c), doğrusal olmayan aktivasyon fonksiyonu (Şekil 2.2.d) ve çıkış bölümlerinden oluşmaktadır. Matematik modellerin yan yana dizilmesi ise giriş ve çıkış noktalarını bir birine bağlayan yapay sinir ağını oluşturmaktadır.

Şekil 2.2.c'de yapay sinir modeline sunulan girdi verilerin üzerinde yapılmış olan işlem şematik olarak gösterilmektedir. Denklem 2.4'te olduğu gibi girdi veriler,  $x$ , ile ağ ağırlıkları olarak ifade edilen  $w$  katsayılar arasında çarpma (iç çarpım) toplama işleminin sonucuna sapma (bias),  $\theta$  değeri de eklenerek  $\alpha$  değeri elde edilmektedir. Bu değer aslında yapay modelin girdi sinyaline gösterdiği tepkidir. Bu tezde  $w$  ve  $\theta$  değerleri derin modelin katsayıları olarak tanımlanmaktadır. Yapay sinirin çıktı verileri yani  $y$  değeri  $\alpha$ ' ya bağlı doğrusal olmayan aktivasyon fonksiyonunun,  $f$ , özelliklerine göre belirlenmektedir.  $y$  değerlerinin hesaplanması Denklem 2.5'ta sunulmaktadır.

$$\alpha = \sum_{i=1}^N w_i x_i + \theta \quad (2.4)$$

$$y = f(\alpha) \quad (2.5)$$

Yapay sinir hücresinde denklem (2.4) doğrusal bir dönüşümü gerçekleştirirken aktivasyon fonksiyonu (Denklem 2.5) ise doğrusal olmayan bir dönüşümü uygulamaktadır. Bu dönüşümlerin ard arda gelmesi bir katman olarak adlandırılmakta ve yapay sinir ağları bu katmanların yan yana dizilmesinden oluşmaktadır. Şekil 2.2.d'de sırasıyla soldan sağa *basamak*, *Gaussian* ve *sigmoid* fonksiyonları üç farklı aktivasyon



Şekil 2.2 (a) Biyolojik sinir hücresi, (b) biyolojik siniri yansıtan matematik sinir modeli, (c) matematik sinir modelin katsayıları ile girdi verilerin çarpma toplama sonrası elde edilen sonucun aktivasyon fonksiyonu ile değerlendirilmesi, (d) farklı aktivasyon fonksiyonları (Şekiller Lippmann (1987), ve van der Baan ve Jutten (2000) 'den alınmıştır

fonksiyonu örneği olarak sunulmaktadır.

Yapay sinir ağının söz konusu dönüşümler sayesinde girdi verilerin değerlerini, boyutunu ve uzamını değiştirebilme potansiyeli farklı problemlerin çözümünde bir yöntem olarak değerlendirilmesine neden olmaktadır. Takip eden tarihi süreçte, söz konusu matematik modele uyumlu olarak Frank Rosenblatt'ın ortaya koyduğu ilk hesaplamalı model olarak tanımlanan algılayıcılar (perceptrons) (Rosenblatt, 1958) bugünkü yapay sinir ağlarının öncüsü olarak kabul edilmektedir (Tappert, 2019).

Rosenblatt'ın perceptronu doğrusal olarak ayrıştırılabilen problemlerde etkin çalışırken, doğrusal olmayan karmaşıklıkta örneğin görüntülerin modellenmesinde yetersiz kalmaktadır. Ancak, bu yapının ardışık katmanlarla birbirine (arada doğrusal olmayan

aktivasyon fonksiyonlarıyla) bağlanmasıyla elde edilen sinir ağı, bu sorunu aşabilmektedir. Araştırmalar, belirtilen şekilde bağlanmış en az bir gizli katmana sahip bir yapay sinir ağı modelinin tanımlanabilecek tüm fonksiyonları modelleyebilecek, evrensel bir yakınsayıcı olduğunu göstermiştir (Cybenko, 1989). Bu bağlamda, giriş ve çıkış katmanlarının arasındaki sinir sayısı arttırılabilmekte ve bu sayı arttırıldıkça elde edilen sinir ağı daha da derinleşmekte ve modelin katsayılarının ( $w$  ve  $\theta$ ) sayısı da artmaktadır. İki'den fazla katman sayısına sahip (en az iki gizli katmanı olan) sinir ağı literatürde derin sinir ağı veya derin model olarak tanımlanmaktadır.

Denklem 2.4'te olduğu gibi derin modelin çıktısını,  $y$  değerini belirleyen katsayıların değiştirilmesi ile yeni çıktılar üretilebilmektedir. Katsayı kümesinin oluşması hedeflenen çıktı değerlerine doğru değiştirilmesini sağlayan süreç *öğrenme* (eğitim) olarak adlandırılmaktadır. Söz konusu katsayıların (ağ ağırlıklarının) belirlenmesinde izlenen yöntem **derin öğrenme** olarak tanımlanmaktadır. Derin modelin eğitim süreci öğrenilmesi planlanan konu ile ilgili uygun veri kümesi alanında gerçekleşmektedir. Statik dağılımın veya lojik bir ilişkinin öğrenilmesi için hazırlanan söz konusu veri seti *eğitim veri kümesi* olarak tanımlanmaktadır. Sadece girdi verileri kullanılarak gerçekleştirilen eğitim süreci deneyimsiz (unsupervised) ve girdi-çıktı(ground truth) şeklinde çift veri setleri kullanılarak tanımlanan eğitim süreci ise deneyimli (supervised) derin öğrenme yöntemi olarak tanımlanmaktadır.

Birinci katmandan başlayarak her bir katmanda, girdiler ile ağ ağırlıklarının iç çarpma toplama işleminin sonucu o katmanın aktivasyon fonksiyonu süzgeçinden geçilerek sıradaki katmana girdi verisi olarak aktarılmaktadır. Deneyimli derin öğrenme yönteminde son katmanın ürettiği veri/veriler derin modelin çıktısı olarak eğitim veri setindeki amaç verisi/verileri ile karşılaştırılarak aradaki fark önceden belirlenmiş uygun bir amaç fonksiyonu (loss function) ile değerlendirilmektedir. Ağırlıklı olarak yanlış enerji veya kayıp fonksiyonu adı ile ifade edilen amaç fonksiyonu ağ tahmininin hatasını modellemektedir.

Hata değerinin, öğrenilen parametrelere göre parçalı türevleri hesaplanarak geri ya-

yılım (backpropagation) algoritması ile tüm katsayı değerlerinin çözüme yönelik değiştirilmesinde kullanılmaktadır (Rumelhart vd., 1986). Bu sırada, amaç fonksiyonu ile elde edilen hatanın azaltılması için katsayıların değişme yönü ve büyüklüğü en-iyileme (optimizasyon) yöntemi ile hesaplanmaktadır. En-iyileme süreci, kayıp fonksiyonundaki azalmanın değişmez hale gelmesine dek tekrarlanmaktadır. Derin modele sunulan tüm eğitim verileri üzerinden geçildiğinde bir *epok* olarak tanımlanmaktadır. İteratif olarak ağ güncellemesi yapılan bu sürecin sonunda, ağın ürettiği çıktılarla, eğitim verisinde amaç verisi olarak hazırlanmış etiketli verilerin arasındaki farkın azalması beklenilmektedir.



### 3. EĞİTİM VERİ KÜMESİ

Eğitilmiş derin modelin farklı yeraltı hız modellerini kabul edilebilir bir hata payı ile kestirebilmesi eğitim aşamasında kullanılan eğitim veri kümesinin sayısından ve kapsamından etkilenmektedir. Yeraltı modeli düşey ve yatay yönde elastik modülleri ve onlara bağlı olarak sismik hız değerleri farklı olan birimlerden ve tabakalardan oluşmaktadır. Yeraltındaki her bir jeolojik tabakanın derinliği, üst ve alt tabakalarla aralarındaki geçiş yüzeylerin şekli, ve kalınlık değerleri değişmektedir. Derin model tabakaların özellikleri ile birlikte hız modelinin tabaka sayısını da öğrenmelidir.

Belirtilen parametrelerin değişim aralığı yeraltı modellerin kapsayabileceği olası aralığını da göstermektedir. Tanımlanan eğitim senaryosu ile eğitim verilerinin kapsadığı alanın örtüşmesi gerekmektedir. Örnek için sadece dört tabakalı hız modelleri ile eğitilen bir derin model iki veya beş tabakalı hız modellerini başarılı bir şekilde kestiremez veya kestirim hata payı dört tabakalı hız modellerinden daha büyük olur. Amaç iki, dört ve beş tabakalı hız modellerinin kestirimi ise eğitim veri kümesi her üç hız modelini de kapsayacak şekilde hazırlanmalıdır.

Eğitimi veri kümesindeki örnek sayısı derin modeldeki katsayıların düşük hata değerleri ile sonuç üretecek yönde değişmesi için yeterli olmalıdır. Yeterli değer hesaplanması için herhangi bir denklem bulunmamaktadır. Ancak söz konusu örnek sayısı eğitilmesi veya çözülmesi hedeflenen problemin karmaşıklık seviyesi ve tasarlanmış olan sinir ağlarının derinliği ve dolayısı ile katsayı kümesinin boyutu ile ilişkili olmaktadır. Problemin karmaşıklık derecesi ve sinir ağlarının derinliği arttıkça eğitim veri kümesindeki örnek sayısı da artmalıdır. Örnek olarak üç ve dört tabakalı hız modellerin kestirimi için planlanmış olan derin öğrenme senaryosunda hazırlanan eğitim veri kümesinin boyutu ve tasarlanmış olan sinir ağlarının derinliği tabaka sayıları üçten dokuza dek değişen hız modellerin kestirimini içeren diğer bir senaryo için yetersiz kalabilmektedir.

Deneyimli derin öğrenme yönteminde sismik veriler kullanılarak hız modellerinin kestirimi için etiketlenmiş girdi-çıkı eğitim veri çiftleri gerekmektedir. Bunun için de her bir hız modelinde, çıkı verileri, hesaplanan sismik verileri girdi olarak kullanılabilir. Böylece girdi ve çıkı olarak etiketlenmiş eğitim veri çifti 'sismik verileri - hız modeli' şeklinde tanımlanabilir. Derin sinir ağıları girdi verilerini ilgili çıkı verilerine dönüştürerek aslında sismik verilere ait hız modellerin kestirilmesini sağlayabilen katsayı kümesini hesaplamış olmaktadır.

Hazırlanmış olan her bir hız modelinde sismik dalga denklemi çözümlenerek sentetik sismik verilerin hesaplanması ile girdi verileri elde edilmektedir. Farklı hız modelleri için hesaplanan sismik verileri bir biri ile farklı olabilmektedir. Ancak sismik dalga denklemi doğrusal bir denklem değildir bu nedenle de hız modelin parametreleri ile sismik verilerin arasında doğrusal olmayan bir ilişki bulunmaktadır. Söz konusu doğrusal olmayan ilişkiyi derin modelin öğrenebilmesi sayı ve kapsam olarak uygun eğitim veri kümesi ile mümkündür..

Bu tez çalışmasında 2B akustik hız modeli amaç veya kesin referans verisi ve sismik atış kesitleri ise girdi verisi sentetik olarak hazırlanmıştır. Diğer bir deyişle eğitim veri kümesi 'sismik atış kesitleri – hız modeli' veri çiftlerinden oluşmaktadır. Her bir hız modeli için yirmi adet sentetik sismik atış kesiti hesaplanmıştır. Eğitim veri kümesinin sayı ve kapsam açısından yeterli olması için binlerce hız modellerin hazırlanma süreci kontrollü ama rastgele seçimler ile ilerleyen otomatik bir algoritma tanımı ile gerçekleştirilmiştir.

Eğitim veri kümesinin öğrenme sürecinde sayı olarak etkisinin araştırılması için farklı sayılardan oluşan eğitim veri setleri hazırlanmıştır. Eğitim için ayrılmış olan tüm verilerden 750, 1500, 3000, 4500 ve 6000 örnek rastgele bir şekilde ama farklı model gruplarının arasında eşit dağılımla ayıklanarak beş farklı eğitim veri kümesi oluşturulmuştur. Kolaylık olsun diye bu beş farklı sayılı eğitim veri kümesi sırayla TD-I, TD-II, TD-III, TD-IV, ve TD-V olarak adlandırılmaktadır. Bütün denemelerde 12000 örnekten oluşan test denektaşı (benchmark) sabit kalmaktadır.

Yeterli sayıda gerçek sismik verilerin elde edilmesi bir taraftan veri toplama işleminin yüksek maliyetli olduğu diğer taraftan ise veri toplayan firmaların gizlilik prensipleri nedeni ile mümkün değildir. Geleneksel olarak kullanılan hız analizi yöntemleri ile hız modellerin ve dolayısı ile etiketlenmiş veri çiftlerin elde edilmesi de zaman alıcı bir işlemdir. Bu sebeple de eğitim ve test veri kümesi sentetik olarak hazırlanmıştır.

Bu bölümde parametreleri farklı olan çok sayıda hız modellerin üretilmesi ve ardından her bir hız modeli için sismik verilerin hesaplanması yer alarak tez kapsamında hazırlanan eğitim veri kümesi anlatılmaktadır. Ayrıca bu tez çalışmaları kapsamında tasarlanmış olan derin modelin farklı veri kümeleri üzerinde de performansının ölçülmesi amacıyla Deng vd. (2022) tarafından açık kaynak olarak yayınlanmış olan OpenFWI veri seti kullanılmıştır. Söz konusu eğitim veri kümelerinin özellikleri bu bölümün üçüncü alt başlığında ele alınmıştır.

### 3.1 Hız Modelleri

Yerküresi üzerinde herhangi iki noktanın yeraltı gerçek hız modeli büyük olasılıkla bire bir aynı değildir. Ancak değişen parametrelerin sayısı ve belki de değişim aralıklarını belirlemek mümkündür. Jeolojik tabaka sayısı, tabakaların hız değerleri, tabakalar arası ara yüzey şekli, tektonik etkenler nedeni ile kırılmalar veya kıvrılmalar, ve tuz domu gibi sokulumlar hız modellerinin söz konusu değişkenleri olarak sıralamak mümkündür. Önceden belirlenmiş olan hız modeli parametreleri için değişim aralıklarından rastgele değerlerin seçilmesi ile ilerleyen bir algoritma aracılığı ile kullanıcı etkisi olmadan çok sayıda farklı hız modelleri üretilebilir. Bu tez çalışması kapsamında sismik hız modelleri benzer yöntem kullanılarak hazırlanmıştır.

Akustik, homojen, ve izotropik ortam varsayımı ile hazırlanmış olan hız modellerinin tabaka arası hız değerleri tüm noktalarda eşit ve tabakaların derinliği artınca hız değeri artmaktadır. Ara yüzeylerin şekli her bir hız modelin gerçek yeraltı haritasına yakınsamasında önemli rol oynamaktadır. Tektonik ve jeolojik etkenler nedeni ile ta-

bakalar arası geçiş sınırları genelde yatay ve düz şeklinde değildir. Eğimli, antiklinal ve senklinal şeklinde kıvrılan, düzensiz dalgalı ve tırtıklı ara yüzeylerden oluşan hız modelleri yeraltı modellerini daha gerçekçi bir şekilde yansıtmaktadır. Bu nedenle de tabakalar arası olası geçiş şekilleri ile ilgili farklı stratigrafi kaynakları araştırılmıştır.

Rastgele hız modellerin oluşturulması için ilk önce değişen parametreler ve onların değişim aralığı seçilmiştir. Söz konusu parametreler ve değişim aralıkları izleyen şekilde sunulmaktadır.

- **Tabaka sayısı:** üretilmiş olan hız modellerinde tabaka sayısı dörtten sekize dek değişmektedir. Hız modelleri tabaka sayısına göre gruplara bölünerek beş farklı hız modeli grubu oluşmaktadır.
- **Tabakaların derinliği ve kalınlığı:** birinci tabakanın üst sınırı modelin yüzeyinden başlayarak belli bir derinliğe indikten sonra ikinci tabakaya geçiş yapılmaktadır. Aynı şekilde ikinciden üçüncü ve diğer tabakalara da her birinin derinlik değerinde ( $d_i$ ) geçiş yapılmaktadır. Böylece her bir tabakanın derinliği ve dolayısı ile bir önceki tabakanın kalınlığı belirlenmiş olmaktadır.
- **Hız değeri:** tabakalar arası hız değeri ( $V_i$ ) derinlikle artarak 1500 - 4000 m/s aralığında değişmektedir.
- **Ara yüzey şekli:** Tektonik hareketler nedeni ile ortaya çıkan antiklinal ve senklinal şeklindeki kıvrılmalar ayrıca sedimanter ortamların özellikleri sebebi ile oluşmuş olan düzensiz, eğimli ve dalgalı ara yüzeyler dikkate alınarak 116 adet ara yüzey fonksiyonunu kapsayan bir ara yüzey veri kümesi hazırlanmıştır. Her bir ara yüzey fonksiyonunun oluşturulması için trigonometrik, logaritmik ve üstel gibi farklı matematik fonksiyonların birleşimi kullanılmıştır. Söz konusu matematik fonksiyonlardan örnek olarak izleyen şekilde üç fonksiyon sunulmaktadır (Denklem 3.1-3.3).

$$f_1 = \sqrt{x} + 5 \log(15x + 1) \sin(0.14x) \cos(0.3x + 20) \quad (3.1)$$

$$f_2 = 0.09x^2 - 3.6 \sin(0.6x + 2) + 5(x + 1.5) - 0.8x \quad (3.2)$$

$$f_3 = e^{(0.1x-0.11)} - 1.5 \sin(x - 6) + 2.1 \frac{x}{x + 3} \quad (3.3)$$

Dikkat edilmesi gereken husus şudur ki hız modellerin ara yüzey şekli sadece 116 adet eğri fonksiyonundan oluşmamaktadır. Algoritma 1’de sunulduğu gibi yazılmış olan otomatik hız modeli üretme sürecinde rastgele olarak seçilen ara yüzey fonksiyonu üst tabakanın (ilk tabaka hariç) ara yüzey fonksiyonu ile toplanarak alt tabakalara inmektedir. Böylece ara yüzey şekilleri rastgele olarak oluşmakta ve sayı olarak da 116 adetden daha büyük olmaktadır. Yukarıda anlatılan parametreler kullanılarak rastgele seçimlerle ilerleyen bir algoritma aracılığı ile beş farklı hız modeli grubu oluşturulmuştur. Bu şekilde elde edilen hız modeller tabakalı modeller olarak tanımlanmaktadır.

Denklem 3.1-3.3’te yazılmış olan fonksiyonlar kullanılarak Şekil 3.1’de iki farklı hız modelinin hazırlanma aşaması örnek olarak gösterilmiştir. Şekil 3.1.a’da ara yüzey şekillerinin oluşturulması için sırayla  $f_1$ ,  $f_2$  ve  $f_3$  ara yüzey fonksiyonlarının seçilmesi varsayılmaktadır. Fonksiyonlar seçildikten sonra sırayla ilk ara yüzey,  $L_1 = f_1$ , ikinci ara yüzey,  $L_2 = f_1 + f_2$ , ve üçüncü arayüzey,  $L_3 = f_1 + f_2 + f_3$ , oluşmaktadır. Son aşamada ise tabakalar arası hız değerleri belirlenmiş aralıklardan rastgele seçilerek hız modeli oluşmuş olmaktadır.

Şekil 3.1.b’de aynı fonksiyonlar ama farklı sırayla  $f_3$ ,  $f_2$  ve  $f_1$ , kullanılmaktadır. Aynı

---

#### **Algoritma 1** Çok Tabakalı Hız Modeli Algoritması

---

**Gereken:** Parametrelerin stokastik seçimi, Toplam tabaka sayısı  $N$

Başlat  $d_{prev} \leftarrow 0$

Başlat  $L_{prev} \leftarrow 0$

**for**  $i = 1$  to  $N$  **do**

$V_i \leftarrow i$ . tabaka için hız değerini seç

$d_i \leftarrow i$ . tabakanın derinliğini seç

$L_i \leftarrow i$ . tabakanın ara yüzünü seç

Başlangıç sınırı  $\leftarrow d_{prev} + L_{prev}$

Bitiş sınırı  $\leftarrow d_{prev} + L_{prev} + d_i + L_i$

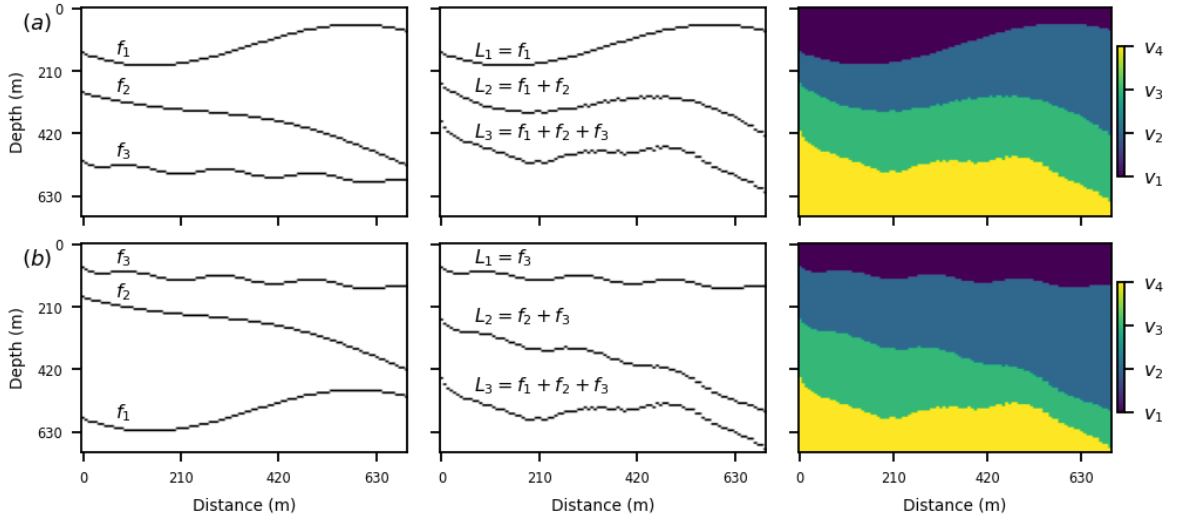
Başlangıç ve Bitiş sınırlarının arasına  $V_i$  değerini ata

Güncelle  $d_{prev} \leftarrow d_{prev} + d_i$

Güncelle  $L_{prev} \leftarrow L_{prev} + L_i$

**end for**

---



Şekil 3.1 Denklem 3.1-3.3'te sunulan fonksiyonlar kullanılarak farklı ara yüzeyler ve hız modellerin oluşturulması

sürecin sonucunda hazırlanmış olan hız modelinin (hız değerleri değiştirilmemiştir) yapı olarak önceki hız modelinden farklı olduğu açık bir şekilde görünmektedir. Ayrıca ortaya çıkan ara yüzey şekilleri kullanılan fonksiyonların şekli ile farklı oldukları da ayırt edilmektedir. Böylece tasarlanmış olan algoritma gerçek anlamda rastgele hız modelleri üretmektedir.

Tektonik hareketler yer kabuğunda sıkışma ve gerilme zonları oluşturarak yeraltı tabakalarının farklı yönlerde, farklı atımlar ve açılar ile kırılmalarına neden olmaktadır. Eğitim veri kümesinde daha gerçekçi hız modellerin yer alması amacı ile hazırlanmış olan hız modeli gruplarına fay eklenerek her bir grubun ayrıca faylı şekli de elde edilmiştir. Fay ekleme işlemi için dört parametre ve değişim aralığı belirtilerek faylı modelleri üreten algoritma hazırlanmıştır. Söz konusu değişkenler izleyen şekilde açıklanmaktadır.

- **Atım boyu:** hız modeldeki tabakaların fay hattı boyunca yer değişme (kayma) miktarını gösteren atım boyu değişim aralığında rastgele olarak seçilmektedir.
- **Açı:** fayların yatay düzlemi ile yapmış olduğu açı 16-31 derece aralığında değişmektedir.

- **Fay türü:** düz veya ters fay olmasına bağlı olarak modelin gerekli kısmı yukarı/aşağı yönde fay hattı eksenini yönünde hareket etmektedir.
- **Yatay konum:** hız modeli içerisinde kırılma hattının yatay konumunun değişmesi farklı hız modellerinde fay hattının farklı yatay konumunda olmasını sağlamaktadır. Yatay konumun değişim aralığı hız modellerin sağ ve sol kenarlarından 140m mesafeyi koruyacak şekilde ayarlanmıştır. Fay hattının yatay konumu modelin bir kenarına yaklaşıncaya açısına göre kenara yakın tarafı gölgede kalabilir diğer bir deyişle o tarafı sismik dalgalarca yeterli düzeyde taranamamaktadır. Bu ise fay ile ilgili sismik verilerde daha az bilginin yer alması anlamına gelmektedir.

Petrol ve doğal gaz aramalarında önemli belirti olarak değerlendirilen tuz domları şekil ve hız değeri açısından etraflarındaki sedimanter birimlerden farklı olmaktadır. Sokulum şeklinde alt tabakalardan yukarıya doğru çıkmış olan tuz domları faylanma veya diğer sebeplerden dolayı tabakaların arasında sokulum birimleri gibi de gözükmektedir. Yukarıya doğru hareket eden tuz domları tabakaları sıkıştırarak ara yüzeylerin deforme edilmesine neden olmaktadır. Beş farklı hız modeli grubunu oluşturan algoritmayı revize ederek en alt tabakadan yukarıya doğru sokulmuş tuz domlu hız modelleri de ayrıca hazırlanmıştır.

Bu modeller oluşturularak yukarıdaki tabakaların tuz domu şekline uygun olarak deforme olması da uygulanmıştır. Tuz domlu hız modellerin hazırlanmasında rastgele olarak seçilmiş olan parametreler izleyen şekilde sunulmuştur.

- **Tuz domu şekli:** alt tabakadan sokulum şeklinde yukarıya doğru çıkan tuz domlarının şekli en az dört Gauss fonksiyonu kullanılarak oluşturulmuştur. Gauss fonksiyonların parametreleri rastgele seçimlerle değiştirilerek farklı şekillerin üretilmesi sağlanmıştır.
- **Hız değeri:** tuz domlarının hız değeri  $4350 - 4550m/s$  aralığında değişmektedir.
- **Yatay konum:** tuz domlarının yatay konumu hız modelin sağ ve sol tarafından

140m mesafe korunacak şekilde deđişmektedir.

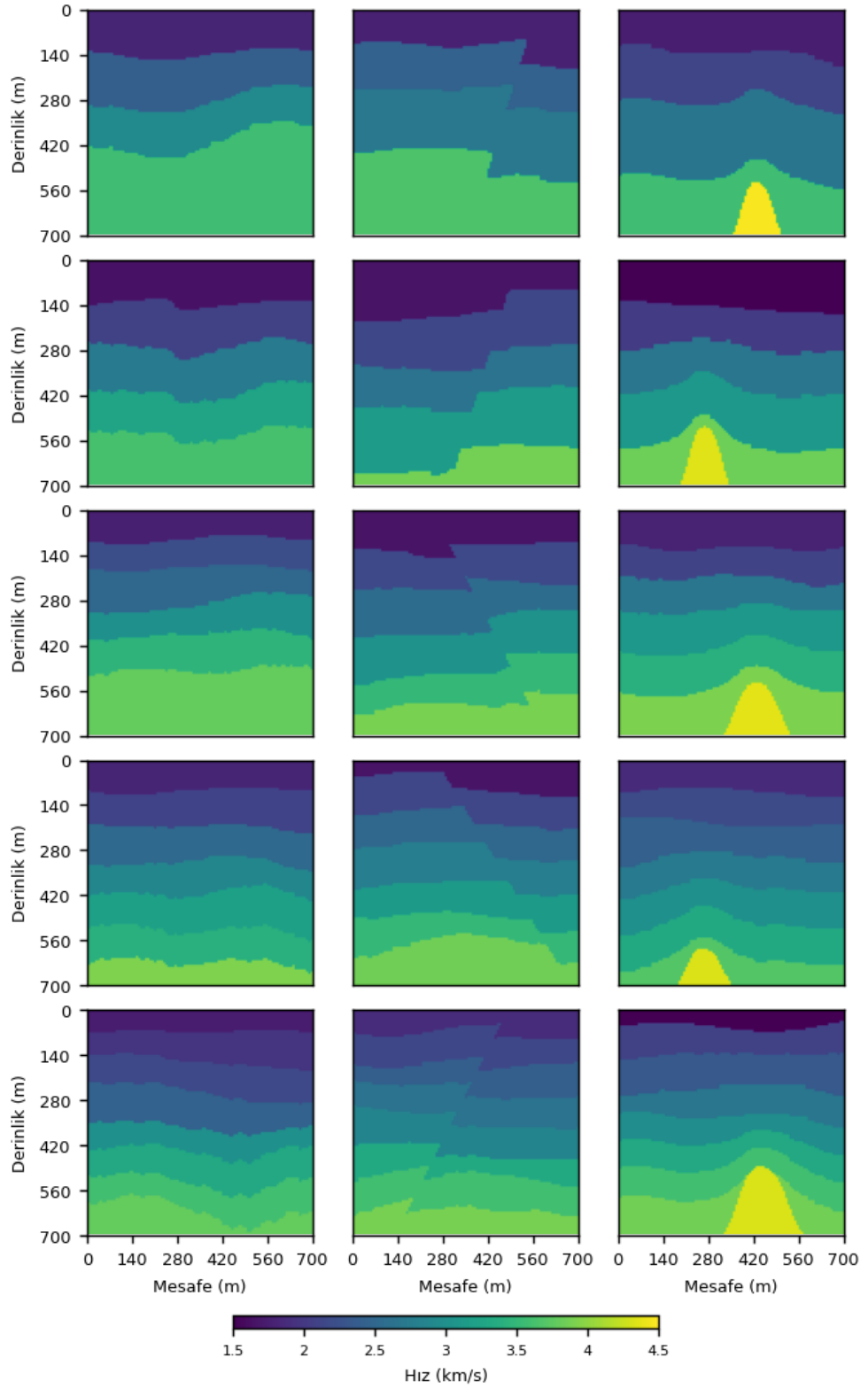
Böylece elde edilmiş olan hız modellerin tabaka sayısı dörtten sekize dek deđişerek beş farklı gruba ayrılmakta ve her bir grup ise sırasıyla *tabakalı modeller*, *faylı modeller* ve *tuz domu içeren modeller* olarak üç ayrı alt gruptan oluşmaktadır. Örnek için üçüncü gruba ait hız modelleri altı tabakalı çoklu tabakalı, altı tabakalı faylı ve altı tabakalı tuz domu içeren modellere ayrılmaktadır. Çoklu tabakalı modeller fay veya tuz domu içermeyen sadece ardı arda dizilmiş tabakalardan oluşmaktadır. Rastgele ve otomatik algoritmalar kullanılarak her bir hız modelinden bin iki yüz adet model oluşturularak toplam 18000 ( $5 \times 3 \times 1200$ ) hız modeli hazırlanmıştır.

Hız modelleri matris olarak deđerlendirilirse  $100 \times 100$  gritten oluşmaktadır. Grid aralıkları  $7m$  olarak hesaplandığı içi her bir hız modelin derinliği  $700m$  ve yatay mesafesi ise  $700m$  olmaktadır.

### 3.2 Sismik Verileri

Arama sismiğinde yüzeyden yüzeye  $2B$  veri toplama tekniğinde alıcılar yeryüzünde düz bir hat boyunca dizilmekte ve yapay kaynakların konumları aynı hattın doğrultusunda veya üzerinde yer almaktadır. Yapay kaynağın yeraltına gönderdiği sismik dalga alanı hız deđerleri farklı olan ara yüzeylerden yansıma, kırılma ve saçılma nedeni ile yeryüzüne geri dönerek alıcılarda zaman serisi şeklinde kaydedilmektedir. Kaynak noktasından alıcılara doğrudan yayılan dalga alanı da bu kayıttta bulunmaktadır. Her bir alıcıda kaydedilen sismik verisi bir iz olarak adlandırılır. Veri toplama hattı boyunca konumlanan alıcıların sırasıyla yan yana dizilmiş izler sismik kesitini oluşturmaktadır. Her bir kaynağın etkisi nedeni ile alıcılarda kaydedilen verilerin oluşturduğu kesit, sismik atış kesiti olarak adlandırılmaktadır.

Hazırlanmış olan hız modellerinde sismik atış kesitlerin hesaplanması için alıcı ve kaynak noktaları hız modellerin yüzeyinde konumlanmıştır. Sayısal olarak yazılırsa alıcıların konumu ilk gridden başlayarak üçer grid aralıklı son gride dek devam et-



Şekil 3.2 Beş grup hız modeline ait örnek modeller. Her bir gruptan tabakalı, faylı ve tuz domu içeren alt gruplardan birer örnek yar almaktadır

miştir. Kaynak noktaları ise üçüncü gridden başlayarak beşer grid aralıkla doksan sekizinci grid son kaynak noktası olarak belirlenmiştir. Böylece hız modellerin yüzeyinde toplam 34 alıcı ve 20 atış/kaynak noktası belirlenmiştir. Diğer bir deęişle her bir hız modeli için yirmi sismik atış kesiti hesaplanmıştır. Hız modellerin toplam sayısı (18000) dikkate alınarak hesaplanan sismik atış kesitlerin toplam sayısı 360,000'e ulaşmaktadır.

Eđitim sürecinde girdi verileri olarak kullanılan sentetik sismik verileri  $2B$  akustik dalga denklemi (Denklem 3.4) kullanılarak hesaplanmıştır. Bu denklemde  $P$  akustik dalga alanını,  $v$  ortamın hız modelini,  $x$  ve  $z$  uzam eksenlerini,  $S$  sismik kaynađını, ve  $t$  ise zaman eksenini simgelemektedir. Denklem 3.4'ün çözümü için farklı sayısal yöntemler bulunmaktadır ancak bu tez çalışmasında sonlu farklar yöntemi kullanılarak zaman ortamında açık çözüm tekniđi ile hesaplanmıştır. Yapay kaynak,  $S$ , için merkezi frekansı  $20Hz$  olan Ricker dalgacıđı kullanılmıştır. Hesaplama algoritması her bir zaman adımında,  $\Delta t$ , dalga alanını tüm grid noktalarında  $(x_i, z_i)$  hesaplayarak alıcı noktalarındaki sismik dalga alanı deđerini kaydetmiştir.

$$\frac{\partial^2 P}{\partial t^2} - v^2 \left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial z^2} \right) = \mathcal{S}(t, x, z) \quad (3.4)$$

Uzam eksenleri,  $x$  ve  $z$ , sırası ile  $\Delta x$  ve  $\Delta z$  örnekleme aralıkları ile sayısallaştırılmıştır. Hesaplamalarda kolaylık olması için bu iki deęişkenin miktarı eşdeđer,  $h$ , olarak belirlenmiştir. Sentetik dalga alanı hesaplamada doğruluk (accuracy) payını çođaltmak ve uzamda saçılma (grid dispersion) deđerini azaltmak için ortamın hız deđer,  $V$ , kaynađın frekansı,  $f_s$ , zaman ve uzam eksenlerinin örnekleme aralıkları,  $h$  ve  $\Delta t$ , sonlu fark çözümünün kararlı olacađı şekilde seçilmelidir (Alford vd. 1974). Hız modellerin minimum hız deđer (1500m/s) dikkate alınarak sayısal hesaplamada dispersiyon etkisini önlemek amacı ile Alford vd. (1974) koşulu uygulanarak grid aralıkları  $7m$  olarak seçilmiştir. Bu hesaplamada kullanılan denklem izleyen şekilde yazılmaktadır.

$$\Delta x = \Delta z = h \leq \frac{V_{min}}{10f_s} \quad (3.5)$$

Maksimum hız değeri ( $4550m/s$ ) ve grid aralığı dikkate alınarak hesaplanan sismik verilerin doğruluk payını artırmak için Lines vd. (1999) koşulu ( Denklem 3.6) kullanılarak zaman ortamında örnekleme aralığı  $1ms$  olarak hesaplanmıştır.

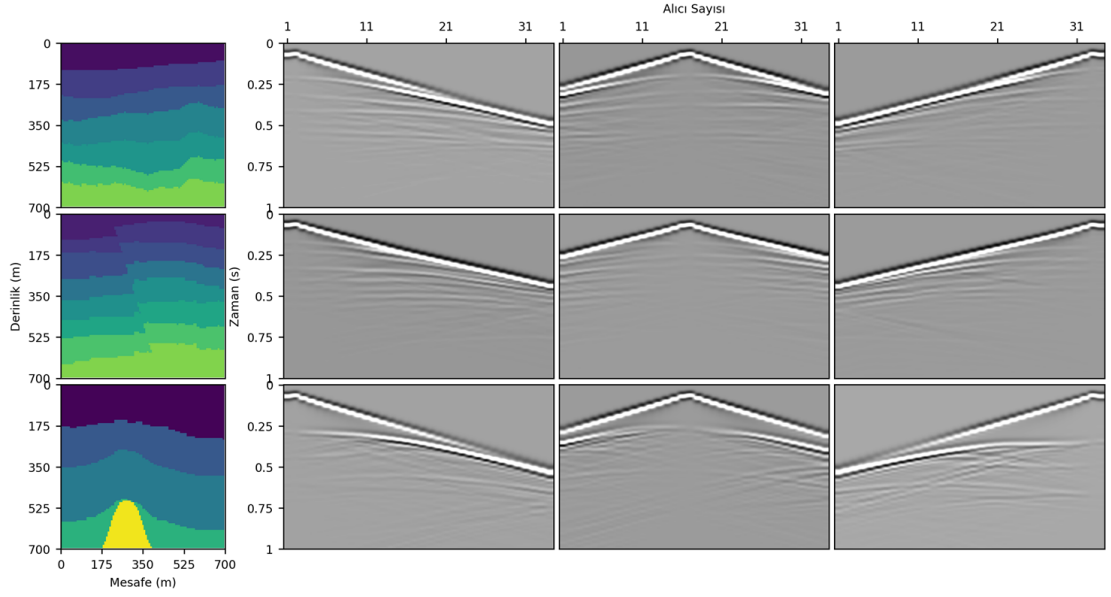
$$\Delta t \leq \frac{h}{\sqrt{2}V_{max}} \quad (3.6)$$

Yazılan koşulların amacı dalga alanının uzam boyutundaki yayılma zamanı,  $\Delta t'$ , diğer bir deyişle  $i$  gridinden  $i + 1$  gridine seyahat süresi, zaman ekseninin örnekleme aralığından,  $\Delta t$ , büyük olmamasını garanti etmektir.

Sınırlı alanda, hız modellerin boyutu, hesaplanan dalga alanı denklemi ile elde edilen sismik kesitlerde sınırlardan istenmeyen geri yansıma dalgaları da bulunmaktadır. Hız modelin sağ, sol, ve alt sınırları ve dört köşesi sırasıyla bir geri yansıma ara yüzeyi ve saçılma noktası gibi hesaplama sürecinde gürültü kaynağı gibi rol oynamaktadır. Hesaplanan sismik kesitlerde söz konusu geri yansıma dalgaların etkisini azaltmak için hız modellerin sağ, sol ve alt kısmına yirmişer grid eklenmiştir. Bununla birlikte hız modelin sınırları ve köşelerine ulaşan sismik dalga alanının zayıflanmasını sağlayan Clayton ve Engquist (1980) sınır sönümlenme koşulu da uygulanmıştır.

Her bir alıcıda sismik izin hesaplanma zamanı bir saniye olarak ayarlanmıştır. Zaman eksenini örnekleme aralığı dikkate alınarak hesaplanan sismik izlerin örnek sayısı toplam 1000 adetten oluşmaktadır. Hesaplamaların sonunda, kaynak sayısı, zaman serisi ve alıcı sayısı olarak her bir hız modeli için  $[20, 1000, 34]$  boyutunda sismik atış kesitleri elde edilmiştir. Diğer bir deyişle eğitim süresinde her bir hız modelin eğitimi için yirmi adet sismik atış kesiti girdi olarak derin modele sunulmaktadır. Benzer şekilde test aşamasında da aynı sayıda sismik atış kesiti kullanılmalıdır.

Şekil 3.3'te rastgele seçilen tabakalı, faylı ve tuz domu içeren hız modellerine ait sismik atış kesitleri çizilmiştir. Sismik kesitleri her bir hız modeli için, sırasıyla birinci, onuncu ve yirincinci kaynak noktasına aittir. Atış kesitlerin arasındaki gözle ayırt edilebilir nitel farklılıklar modeller arasındaki yapısal ve hız dağılımındaki farktan



Şekil 3.3 Sismik hız modelleri ve örnek atış kesitleri

kaynaklanmaktadır.

### 3.3 Sismik Verilere Rastgele ve Tutarlı Gürültülerin Eklenmesi

Arazide toplanan gerçek verileri sayısal olarak hesaplanan sentetik verilerden farklı kılan değişkenlerden biri gerçek verilerde bulunan rastgele ve tutarlı (coherent) gürültülerdir. Hesaplanan sismik atış kesitlerine her iki gürültü türü eklenerek elde edilen gürültülü kesitler gerçek verilere bir adım daha yakınlştırılmıştır. Böylece hız modelleri sabit kalarak gürültülü sismik kesitlerle yeni bir eğitim veri kümesi de elde edilmiştir. Diğer bir deyişle eğitim veri kümesi gürültüsüz ve gürültülü olarak iki farklı veri setine ayrılmaktadır.

Rastgele gürültüler otomatik olarak her bir atış kesiti için rastgele bir şekilde oluşturulmuştur. Tasarlanan algoritmada ilk önce ortalama değeri sıfır ama standart dağılımı rastgele seçilen  $1000 \times 34$  boyutunda bir beyaz gürültü (white noise) kesit oluşturulmuştur. İkinci aşamada beyaz gürültülü kesitin rastgele seçilerek değerlerinin büyük bölümü sıfıra eşitlenmiştir. Son aşamada frekans değeri  $13 - 17Hz$  aralığında rastgele olarak seçilmiş olan sinüs fonksiyonu ile hazırlanan kesit arasında

evirişim işlemi gerçekleştirilerek rastgele gürültü kesiti hazırlanmıştır. Her bir atış kesiti için tekrarlanan bu otomatik işlemin sonunda oluşan rastgele gürültü kesitlerin arasında iki kesitin bire bir aynı olma olasılığı neredeyse sıfıra yakındır.

Tutarlı gürültüler geometrik olarak sismik kesitlerde belirli bir düzenle ortaya çıkmaktadır. Atış kesitlerinde daha düşük hızla ortaya çıkan yüzey dalgaları tutarlı gürültülerin açık bir örneğidir (Chapman 2004). Bu çalışma kapsamında genel olarak yüzey dalgalarının özelliklerinden esinlenerek tutarlı gürültüler oluşturulmuştur. Her bir hız modeli için otomatik olarak hazırlanan tutarlı gürültülerin algoritması ve kullanılan parametreler izleyen şekilde açıklanmaktadır.

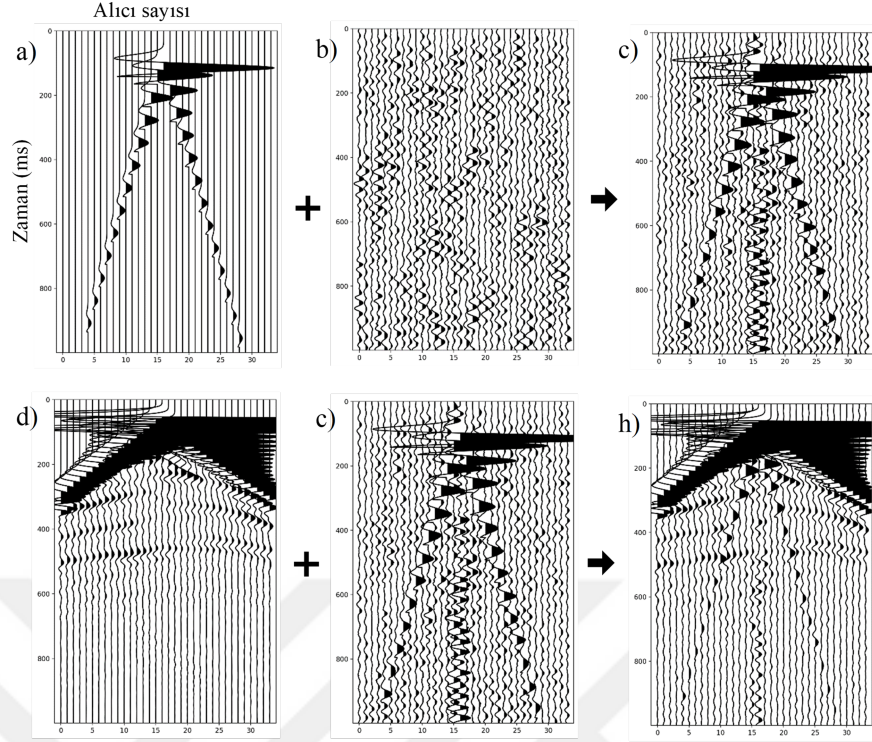
- İlk aşamada her bir hız modeli için tutarlı gürültünün hız değeri ( $v_n$ ) rastgele bir şekilde  $250 - 450m/s$  aralığından seçilerek atış noktalarından ( $S_j$ ) tüm alıcılara ( $R_i$ ) doğrudan seyir ettiği varsayılan gürültü dalganın varış zamanı ( $t_{ji}$ ) hesaplanmaktadır.

$$t_{ji} = \frac{S_j - R_i}{v_n} \quad (3.7)$$

Burada  $j$  değişkeni  $1 - 20$ ,  $i$  değişkeni  $1 - 34$  ve  $n$  değişkeni ise  $1 - 18000$  (hız modellerin sayısı) dek değişmektedir. Böylece her bir hız modeli için atış sayısı kadar sadece varış zamanlarını içeren kesitler oluşturulmaktadır. Dalga yayılımı sönümlenmesi dikkate alınarak varış zamanları  $t_{ji}^2$  bölünerek  $1/t_{ji}$  kesiti elde edilmektedir.

- İkinci aşamada elde edilen kesitler ile merkezi frekans değeri  $8Hz$  ile  $17Hz$  aralığından rastgele olarak seçilmiş olan Ricker dalgacığı arasında evirişim işlemi uygulanmaktadır.

Rastgele ve tutarlı gürültü kesitleri hazırlandıktan sonra onlara denk gelen atış kesitleri ile toplanarak gürültülü atış kesitleri elde edilmiştir. Bu süreç Şekil 3.4'te örnek bir atış kesiti için sunulmaktadır. Gürültülerin genliği hesaplanan yansıma dalgalarının genlik boyutunun aralığında olduğu için yansıma hiperbolleri özellikle ikinci ve



Şekil 3.4 Hesaplanan sismik atış kesitine tutarlı ve rastgele gürültülerin eklenmesi. (a) Tutarlı gürültü, (b) Rastgele gürültü, (c) Gürültülerin toplamı, (d) Gürültüsüz atış kesiti ve (h) Gürültülü atış kesiti

sonraki ara yüzeylerden gelenleri daha çok etkilemektedir.

Hız modelinin en alt tabakasında sokulum şeklinde yer alan tuz domları sismik atış kesitlerinde düşük genlikli saçılma hiperbollerini oluşturmaktadır. Atış kesitlerine eklenen gürültüler söz konusu hiperbollerin genliğini bastırarak geometrik düzenin bozulmasına da neden olmaktadır. Aynı şekilde gürültüler faylı hız modellerinde de fay hattının neden olduğu saçılmaları veya yansıma hiperbollerinde ortaya çıkan kaymaları etkileyerek sismik kesitlerdeki fay izlerini sönmülemektedir.

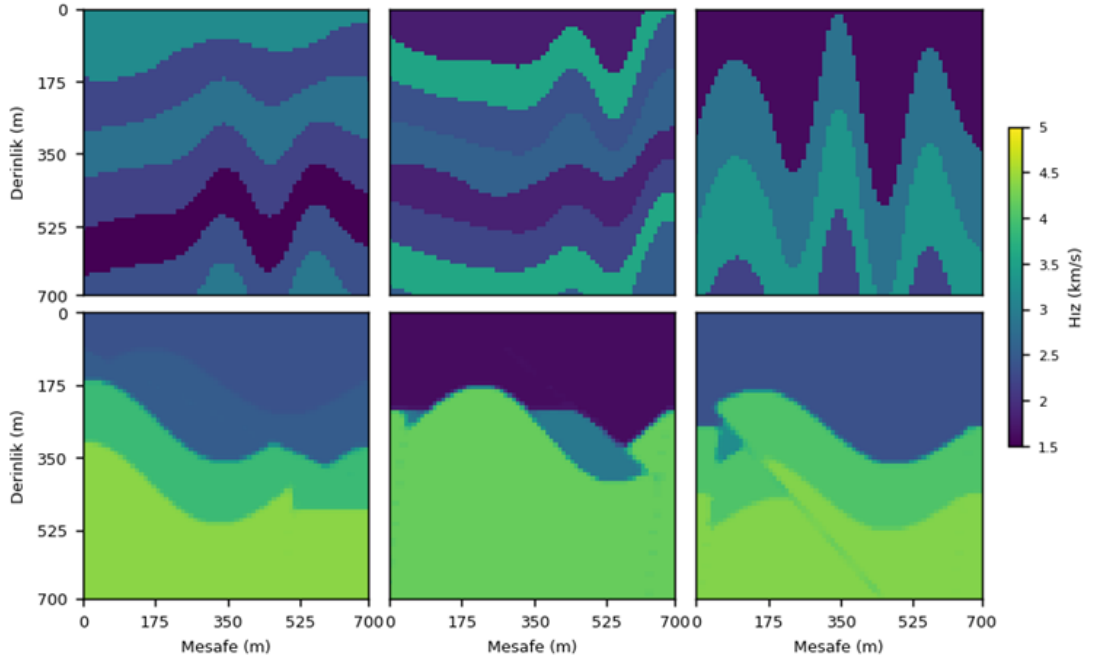
### 3.4 OpenFWI Eğitim Veri Kümesi

Derin Öğrenme çalışmalarında kullanılması amacı ile açık kaynak olarak yayımlanan OpenFWI (Deng vd., 2022) veri kümesi 'sismik atış kesitleri - hız modeli' eğitim veri çiftlerinden oluşmaktadır. Hız modellerinin boyutu, jeolojik yapısı ve karmaşıklık se-

viyesine göre 12 farklı alt gruba ayrılmış olan OpenFWI veri kümesinin *CurveVel-B* ve *CurveFault-A* adlı iki alt veri kümesi bu tez çalışmalarında kullanılmıştır. Böylece SVInvNet derin modelin farklı eğitim veri kümesinde performansı ölçülmüştür.

Eğimli ve dalgalı ara yüzeylerden oluşan çok tabakalı CurveVel-B grubu sınıfının en karmaşık veri kümesi olarak tanımlanmaktadır. CurveVel-B grubu eğitim ve test veri seti olarak sırası ile 24 ve 6 bin hız modeli örneğini kapsamaktadır. Fay hattı içeren çok tabakalı CurveFault-A grubu da eğimli ve kıvrımlı ara yüzeylerden oluşmaktadır. Bu veri kümesi ise 48 bin eğitim ve 6 bin test veri çiftinden oluşmaktadır. İki gruba ait örnek hız modelleri Şekil 3.5'te sunulmaktadır.

Bu tez kapsamında hazırlanmış olan veri kümesinden farklı olarak OpenFWI'a ait bu iki veri grubundaki girdi ve çıktı verilerinin boyutu sırayla  $5 \times 1000 \times 70$  (kaynak sayısı, zaman örnekleri, alıcı sayısı) ve  $70 \times 70$  grid (hız modelin derinliği ve genişliği) şeklinde ayarlanmıştır. Hız modellerin yüzeyinde alıcıları her bir grid noktasında, ve beş atış noktası ise eşit aralıklarda konumlanarak sismik veriler hesaplanmıştır.



Şekil 3.5 OpenFWI veri kümesine ait örnek hız modeller. İlk satırda CurveVel-B ikinci satırda ise CurveFault-A veri setlerine ait örnekler yer almaktadır

Örnekleme aralığı zaman eksenini için  $4m/s$  ve uzam eksenleri için  $10m$  olarak hesaplanmıştır. Tabakalar arası hız değerleri CurveVel-B grubu için  $1500 - 4500m/s$  ve CurveFault-A grubu için de  $1500 - 5000m/s$  olarak ayarlanmıştır.

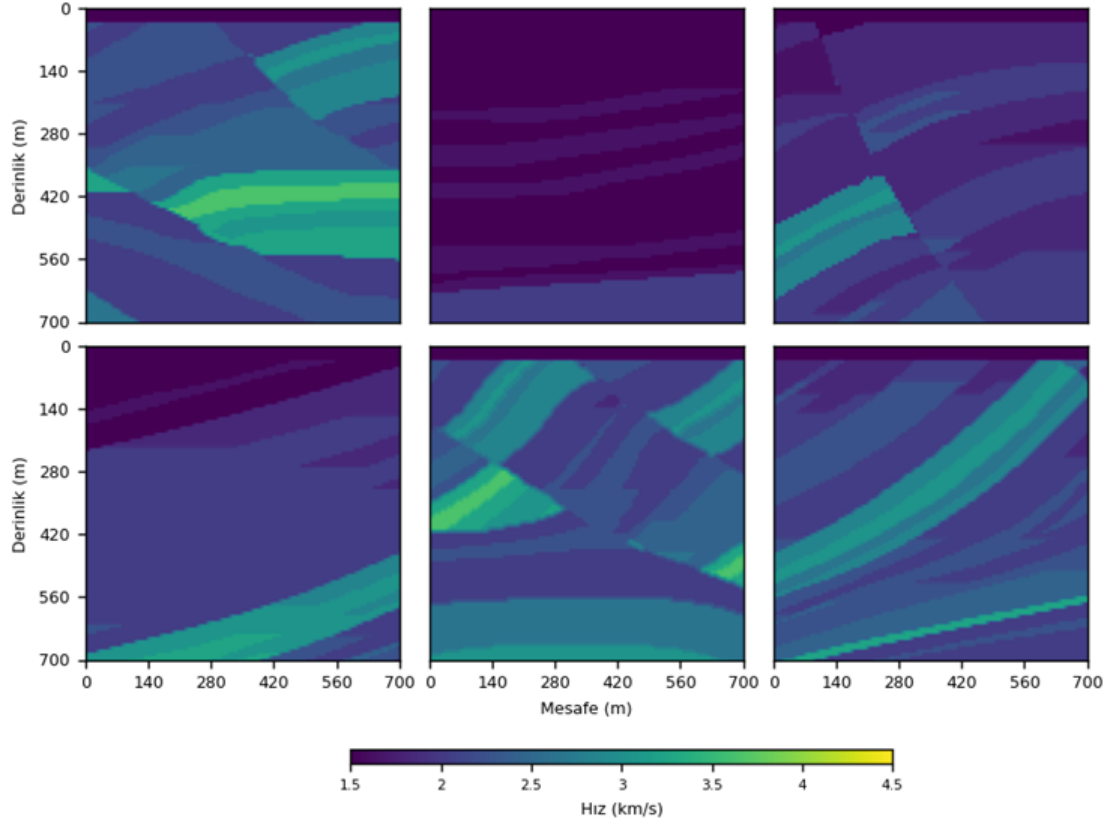
OpenFWI veri kümesinde hız modellerin hız değerleri her zaman derinlikle artmakta ve düşük hızlı tabakalar farklı derinliklerde ortaya çıkarak hız modellerinin karmaşıklık seviyesini bir adım daha zorlaştırmaktadır. Buna ek olarak ara yüzeyler özellikle CurveVel-B grubunda daha çok dalgalı ve kıvrımlı eğrilerle oluşturulmuştur. Bu konu bazen çok abartılı ara yüzeylerin ortaya çıkmasına neden olmakta ancak çok aktif tektonik zonların oluşturabileceği modeller olarak düşünülebilir.

### 3.5 Marmousi Tabanlı Alt Modeller Kümesi

Yeraltı yapılarının karmaşıklığını daha gerçekçi bir şekilde temsil eden Marmousi hız modeli arama sismik alanında örnek hız modeli olarak sıkça kullanılmaktadır. Marmousi modeli, Angola'daki Cuanza Havzası'nda yer alan Kuzey Quenguela Çöküntüsü boyunca uzanan 2B bir sismik profilinden esinlenerek oluşturulmuştur. Bu sentetik model karmaşık hız değişimleri, dik eğimli tabakalar ve yoğun fay zonları gibi yeraltı yapılarını içermektedir.

Marmousi hız modelinin boyutu matris olarak  $751 \times 2301$  noktadan ibarettir. Bu nedenle de bu çalışma kapsamında hazırlanan eğitim veri kümesi veya OpenFWI'daki veri kümeleri ile eğitilen derin modeller test aşamasında Marmousi hız modelini doğru bir şekilde tahmin etmeleri mümkün değildir. Önceki bölümlerde anlatıldığı gibi hazırlanan eğitim veri kümesi  $100 \times 100$  ve OpenFWI ise  $70 \times 70$  gritten oluşmaktadır. Ancak Marmousi hız modelinin farklı bölgelerinden küçük parçalar (alt modeller) seçilerek uygun boyuta ölçeklenen bir Marmousi tabanlı hız modeli kümesinin oluşturulması mümkündür.

Bu amaçla,  $200 \times 200$  boyutunda bir şablon kullanılarak Marmousi hız modeli sistematik bir şekilde belli adım boyu ile taranıp toplam 407 alt model oluşturulmuştur.



Şekil 3.6 Marmousi hız modelinden alınan hız modelleri veri kümesine ait örnek modeller

Elde edilen alt modellerin her birinde düşey yöndeki hız kontrastının en az  $200m/s$  olmasına dikkat edilmiştir. Daha sonra bu modeller  $100 \times 100$  boyutlarına indirgenmiştir (downsample). Hazırlanan 407 hız modelinden 23 model test verisi için rastgele şekilde ayrılmıştır. Geri kalan 384 model kullanılarak eğitim aşamasının gerçekleştirilmesi başarılı sonuçların ortaya çıkması için yetersiz kalmaktadır. Bu yetersizliği gidermek için geri kalan 384 model TD-V veri seti ile birleştirilerek Marmousi alt modellerini de içeren yeni bir eğitim veri kümesi hazırlanmıştır.

Marmousi hız modelinden alınan hız modelleri veri kümesine ait örnek modeller Şekil 3.6'da sunulmuştur. Bu alt hız modelleri Marmousi modelin tüm yapısını içermemekte ancak her bir alt model Marmousi modelin bir kısmını yansıtmaktadır. İç içe girmiş ince tabakalar, yoğun fay zonları, derinlikle artmayan hız değerleri ve tabakaların yatay yönde değişmesi söz konusu hız modellerini önceki veri kümeleri ile farklı kılan özelliklerdir.

#### 4. DERİN MODELİN TASARIMI

Bir problemin çözümü için kullanılan derin öğrenme yönteminin eğitim ve test sürecindeki başarı payı derin modelin tasarımına ve eğitim veri kümesinin sayısına ve kapsamına bağlıdır. Eğitim süreci bittikten sonra elde edilmiş (ağırlık veri kümesi hesaplanmış) derin modelin daha önce görmediği girdiler için de beklenen çıktıyı sunabilmesi, yani genelleme yeteneği yüksek olacak şekilde yapılandırılmış olması esas hedeftir. Derin modelin mimarisi katman sayısı, katmanlardaki yapay sinir (chanel) sayısı, kullanılan yapay sinir çekirdeği, ve katmanlar arası bağlantı düzeni olarak bu hedefe yönelik tasarlanmalıdır.

Ele alınan problemin karmaşıklık seviyesi ve eğitim veri kümesinin sayısı derin modelin tasarımını etkileyen iki önemli etmen olarak tanımlanmaktadır. Problemin karmaşıklık seviyesi artınca derin modelin gizli katman sayısı da artmalıdır. Katman sayısının artması eğitim sürecinde hesaplanması gereken derin modeldeki kat sayıların artması ve karşılığında eğitim veri kümesinin de sayı olarak yeterli olması anlamına gelmektedir.

Optimizasyon süreci, derin modellerde daha çok ve kapsamlı eğitim veri çifti ile daha yüksek başarı gösterme potansiyelindedir. Bu bağlamda, en-iyileme sürecinde karmaşık bazı dengelerin ve düzenleştirme prosedürlerinin takibi önemli olmaktadır. Bu tez çalışması kapsamında yapılmış olan çalışmalarda bu özelliğin sağlandığına ilişkin kontrollü deney ve gözlemler uygulanmıştır.

Tez kapsamında kullanılan mimariler, ağırlıklı olarak oto-kodlayıcı (auto encoder) yapıya uygun ve içinde  $2B$  ve  $3B$  CNN katmanları olan mimarileri içermektedir. Sismik hız modeli ters çözümünü gerçekleştirebilen ve genelleme potansiyeli yüksek olan uygun derin modelin tasarımı için oto-kodlayıcı ve çoklu bağlantılı (dense connection) gibi farklı mimariler incelenerek SVInvNet adı ile tanımlanan özgün derin model mimarisi tasarlanmıştır.

Bu bölümde derin modellerin tasarımı ile ilgili gerekli bilgilere yer verilmektedir. Bölümün devamında Wu vd. (2018) tarafından yayınlanan ve bu tez çalışmasında temel model olarak karşılaştırma amacıyla kullanılan InversionNet derin modelin tasarımı sunulmaktadır. En son alt bölümün altında ise SInvNet derin modelin tasarımı anlatılmaktadır. Eğitim süresi, kullanılan hiperparametreler ve eğitim ve test aşamasında elde edilen sonuçlar son alt bölümde yer almaktadır.

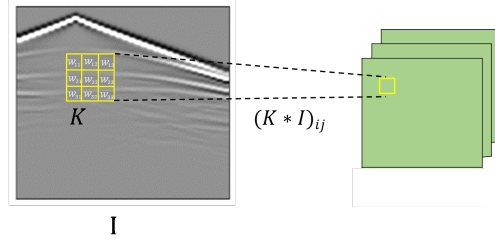
#### 4.1 Evrişimli sinir ağları (CNN'ler)

Günümüzde farklı türde pek çok derin ağ mimarisi, farklı yapısal özellikleri ile değişik problemlerin çözümünde kullanılmaktadır (Shrestha ve Mahmood 219, Dong vd. 2021). Girdi verilerin üzerinde hareket ederek lokal evrişim (konvolüsyon) işlemi ile imge/ingelerden öznitelik çıkarmayı öğrenebilen yapay sinir çekirdekleri ard arda dizilince kademeli olarak çok sayıda farklı özelliğe ait filtre içeren katmanları oluşturmakta ve evrişimli sinir ağları (CNN'ler) olarak tanımlanmaktadır (LeCun, 1989, Fukushima, 1980). Çıkış katmanında tam bağlı sinir ağı kullanılarak CNN mimarileri sınıflandırıcı problemlerde görüntü işleme alanında olduğu gibi başarılı sonuçlar sunmuştur. CNN tasarımını kullanan derin model varyasyonları arasında *AlexNet*, *Inception*, *ResNet* ve *VGG* gibi iyi bilinen modeller yer almaktadır.

Şekil 4.1'de  $2B$  bir CNN çekirdeği ile girdi verinin (örneğin bir sismik atış kesiti) arasındaki evrişim işlemi ve onun sonucunda öznitelik değerlerinin hesaplanması şematik olarak gösterilmektedir. Bu görselin matematik ifadesi Denklem 4.1'de sunulmaktadır. Bu bir evrişim denklemdir ve farklı şekillerde de yazılabilir. Burada Goodfellow vd. (2016) kullandığı şekilde yazılmıştır.

$$S(i, j) = (K * I)(i, j) = \sum_p \sum_q I(i - p, j - q)K(p, q) \quad (4.1)$$

Burada  $S$  öznitelik veya çıktı veriyi,  $I$  girdi veriyi ve  $K$  ise  $(p \times q)$  boyutlu CNN çekirdeğini simgelemektedir.



Şekil 4.1  $3 \times 3$  boyutlu yapay sinir çekirdiği ile girdi verinin arasındaki evrişim işleminin şematik görseli

CNN çekirdeği ( $K$ ) girdi verisinin tamamında gezdirilerek özneliğin ( $S$ ) tüm değerleri hesaplanmaktadır. CNN çekirdeğin katsayıları ( $W_{pq}$ ) değişerek (ilk katmanda girdi verinin değerleri değişmemektedir) özneliğin değerleri de değişmektedir. Böylece evrişimli sinir ağların katmanlarındaki çekirdek sayısı kadar girdi verisi ayrıca filtrelenerek öznelik çıktıları hesaplanmaktadır.

Deneyimli derin öğrenme yönteminde çıkış katmanından geriye doğru gelen bilgilere göre her bir CNN çekirdeğinin katsayıları değişmektedir. Bu değişim derin modelin öğrenilmesi hedeflenen konuyla ilgili özneliklerin çıkarılması yönünde gerçekleşmektedir. Genelde  $2B$  CNN çekirdekleri  $3 \times 3$ ,  $5 \times 5$  ve  $7 \times 7$  olarak seçilmektedir. Tabii  $3B$  çekirdeklerin üçüncü boyutu girdinin boyutuna göre, eşdeğer veya daha küçük, seçilmektedir.

CNN çekirdekleri önceden belirlenen kayma mesafesi veya *adım boyu* ile imgelerin üzerinde hareket etmektedir. Her yeni konumda evrişim işlemi sonucunda özneliğe ait sadece bir değer hesaplanmaktadır. Bu ise boyut olarak özneliğin girdi verisinden daha küçük olması anlamına gelmektedir. Çekirdeğin boyutu ve adım boyu büyüyünce boyut farkı daha da büyük olmaktadır. Özneliğin veya çıktı verilerin boyutunun korunması amacı ile evrişim işleminin esnasında girdi verinin etrafı sıfır değerleri ile çerçevesiz (padding). Bu işlem Türkçe literatürde *dolgu* olarak da adlanmaktadır. Çıktı verinin boyutu ile ifade edilen parametrelerin arasındaki ilişki denklem 4.2'de sunulmaktadır.

$$D_{\text{BktB}} = \frac{D_{\text{girdi}} + 2p - K}{s} + 1 \quad (4.2)$$

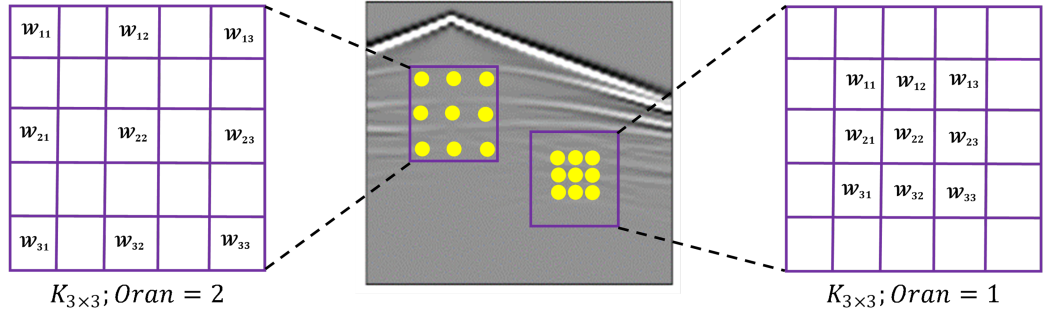
Burada  $D_{\text{BktB}}$  ve  $D_{\text{girdi}}$  sırayla çıktı/öznitelik ve girdi boyutunu,  $p$  dolgu değerini ve  $s$  ise adım boyunu simgelemektedir.

Evrişimli sinir ağlarının her bir katmanı boyutundan bağımsız olarak birden fazla evrişimli çekirdeklerden oluşmaktadır. Kanal (chanel) olarak adlandırılan bu sayı genelde ikinin katsayıları şeklinde örneğin 16, 32, 64 ve 128 olarak seçilmektedir. Katmanların boyutu özniteliklerin boyutu ve çekirdek veya filtrelerin sayısından oluşmaktadır. Örneğin çekirdek sayısı  $c$  olup  $m \times n$  boyutlu öznitelikler üreten bir katmanın boyutu  $m \times n \times c$  olarak literatürde tanımlanmaktadır.

Şekil 4.1'de gösterildiği gibi yapay sinir çekirdeği,  $K$ , imgenin,  $I$ , üzerinde gezdirilince her adımda imge öğelerinin bir kısmını (boyutuna göre örneğin burada  $3 \times 3$ 'lük alan) kapsamaktadır (receptive field). Önceki satırlarda anlatıldığı gibi çekirdeğin katsayıları ile öğelerin arasında evrişim işlemi gerçekleştirilerek öznitelikler çıkarılabilmektedir. Ancak her bölgede kapsadığı öğelerin ortalamasını hesaplayarak veya en büyük olanını seçerek öznitelik haritası oluşturan CNN çekirdekleri de tanımlanmaktadır. *Havuzlama* (pooling) sonucunda öznitelik çıkaran bu çekirdekler kullandıkları işleme göre **ortalama havuzlama** (average pooling) ve **maksimum havuzlama** (max pooling) olarak ayrılmaktadır.

Havuzlama katmanı genellikle bir evrişim katmanından sonra yerleştirilip evrişimden elde edilen özniteliklerin boyutunu küçülterek daha kompakt bir şekilde temsil edilmesine olanak sağlar. Böylece öğrenme sürecindeki hesaplama maliyeti düşmüştür. Ayrıca derin modelin girdi verisindeki ufak kaymalardan etkilenmesini azaltarak (örneğin, görüntüdeki pozisyon değişiklikleri) eğitim sürecini gürbüz (robust) hale getirir.

CNN çekirdeğindeki katsayıların arasına sıfır değerleri eklenerek uygulanan evrişim sonucunda katsayıların ( $W_{pq}$ ) sayısı artmadan imgenin daha geniş bir bölgesi evri-



Şekil 4.2 Genişleme oranı farklı olan  $3 \times 3$  boyutlu CNN çekirdekleri ile imgenin arasındaki evrişim işleminin görseli

şim işlemine dahil edilmiş olmaktadır. Tabii sıfır değerlerine denk gelen söz konusu bölgeye ait öğeler etkisiz hale gelmektedir. **Genişletilmiş evrişim** (atrous convolution) olarak tanımlanan bu yöntemde genişleme oranına göre CNN çekirdeğine sıfır değerleri eklenmektedir. Aslında standart evrişim işlemi genişleme oranı bir olan genişletilmiş evrişimin özel türü olarak değerlendirilebilir.

Şekil 4.2’de genişleme oranı bir ve iki olan  $3 \times 3$  boyutlu iki CNN çekirdeği ile bir imgenin (burada sismik atış kesiti) arasındaki evrişim işlemi şematik olarak çizilmektedir. İmgenin üzerinde çizilmiş olan sarı noktalar çekirdekteki katsayılarla imgenin arasında gerçekleşmiş olan evrişim konumlarını göstermektedir. Her iki oran değerinde de evrişim nokta sayısı sabit kalmakta sadece genişleme oranı iki olan çekirdeğin evrişim noktaları  $5 \times 5$  boyutlu bir çekirdeğin kapsayacağı alanda seyrek olarak dağılmaktadır. Genişleme oranı derin modelin mimarisine göre belirlenmektedir.

#### 4.1.1 Oto-kodlayıcı ağ

Oto-kodlayıcı evrişimli derin modeller tasarımları itibari ile farklı boyutlarda ve uzamlarda olan girdi ve çıktı verilerini birbiri ile eşleştirebilmektedir (Cho vd., 2014a, Goodfellow vd., 2016). Diğer bir deyişle eğitim veri kümesindeki eğitim veri çiftinin aynı boyutta veya alanda olmasına gerek yoktur. Oto-kodlayıcı ağların mimarisi çıktı veriyi üretebilen birim için gerekli bilgileri girdi veri/verilerinden öğrenebilen

birim olarak iki koldan oluşmaktadır. Söz konusu bölümler izleyen satırlarda ayrıca anlatılmaktadır.

- **Kodlayıcı ağ:** Girdi verilerinden çıkarılan öznitelikler kodlayıcı birimin katmanlarında ilerleyerek boyutları küçülmemekte ama sayıları artmaktadır. Bu birimin katmanlarında girdi verinin içerdiği bilgilerin farklı ölçeklerde (low and high level) çıkarılması hedeflenmektedir. Diğer bir deyişle farklı boyutlarda taranan girdi verilerindeki genel ve özel detaylar çok sayıda kullanılan sinir ağları filtreleri ile çıkarılmaktadır.
- **Kod çözücü ağ:** Kodlayıcı biriminden gelen öznitelikler birleştirilerek amaç verisi/verileri ile aynı boyutta olan çıkış katmandaki kestirimin oluşturulması kod çözücü ağında gerçekleştirilmektedir. Bu birim aynı zamanda doğru bir kestirim için gerekli bilgileri çıkış katmanından alıp kodlayıcı ağa geri ileterek tüm modelin başarılı olmasını sağlamaktadır.

Şekil 4.3'te oto-kodlayıcı bir model ve kodlayıcı - kod çözücü birimleri şematik olarak çizilmektedir. Bu şekilde girdi ve çıktı verilerin boyut ve alan olarak farklı olduklarını göstermek için farklı boyutlarda ve renklerde tasarlanmıştır. Kodlayıcı biriminde katman sayısı artınca özniteliklerin boyutu ( $m_l \times n_l$ ) küçülmemekte ( $m_l \times n_l < m_{l-1} \times n_{l-1}$ ) diğer taraftan ise çekirdek veya kanal sayısı ( $c_l$ ) artmaktadır. Simetrik olarak tasarlanmış olan şekil 4.3'te kod çözücü biriminde özniteliklerin boyutu giderek büyümekte ve kanal sayısı ise kodlayıcı birimin tersine giderek küçülmektedir ( $m_{l-1} \times n_{l-1} < m_l \times n_l$  ve  $c_l < c_{l-1}$ ).

Katmanlardaki özniteliklerin boyutu CNN çekirdek boyutu, adım boyu ve dolgu değişkenleri, Denklem 4.2'de gösterildiği gibi, kullanılarak ayarlanabilmektedir. Kodlayıcı birimin son katmanı derin modelin boyut olarak en küçük katmana bağlanarak çıkardığı bilgileri diğer birime iletmektedir. *Kod ya gizli vektör ya da darboğaz* olarak adlandırılan bu katmanın boyutu tasarlanan mimariye göre belirlenmiş olmaktadır. Bu katman Şekil 4.3'te kodlayıcı ve kod çözücü birimlerin ortasında  $m \times n \times c$  olarak çizilmiştir. Bazı araştırmacılar darboğaz katmanını hem kodlayıcı hem de kod

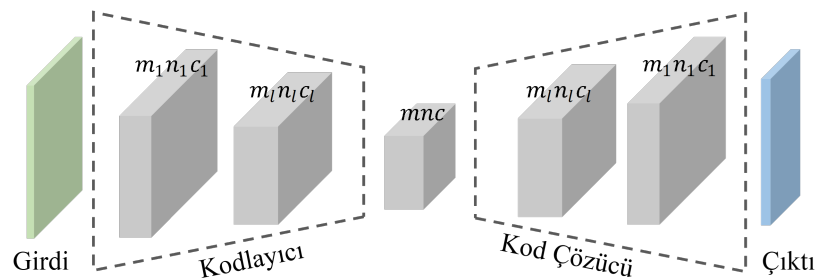
çözücü birimlerle bir arada çizmektedirler.

Kod çözücü biriminde özniteliklerin boyutu iki yöntemle büyütülmektedir. İlk yöntemde evrişim işleminin tersi yani dekonvolüsyon çekirdekleri kullanılmaktadır. Diğer yöntemde ise üst örnekleme (upsample) fonksiyonu evrişim çekirdeği ile birlikte kullanılmaktadır. Bu yöntemde ilk önce özneliğin boyutu büyütülür ardından da evrişim çekirdeği uygulanarak diğer katman/katmanlara iletilir.

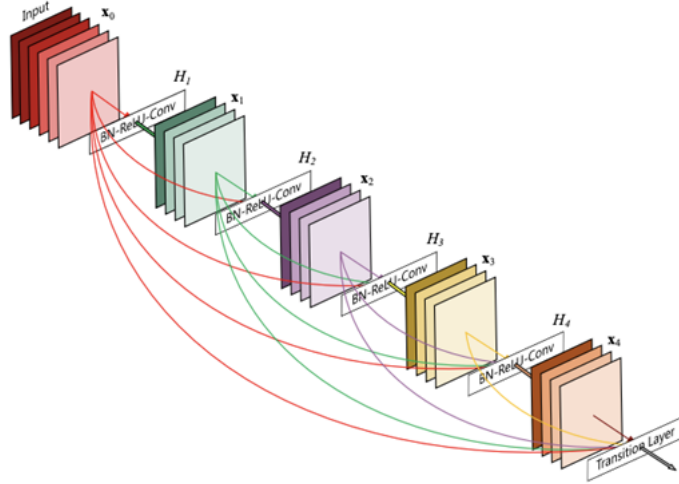
Klasik bakış açısı ile tasarlanan Şekil 4.3'teki derin modelde kod çözücü birimi sadece kod katmanındaki bilgilerle beslenmektedir. Ancak kod çözücü için böyle bir sınırlandırma geçerli değil ve kodlayıcı birimin her bir katmanından bilgi olarak beslenebilmektedir. Bunun bariz örneği literatürde iyi tanınmış olan *U-Net* derin modelidir. Öğrenme sürecinde bilgilerin ileriye (girdi  $\rightarrow$  çıktı) ve geriye doğru (girdi  $\leftarrow$  çıktı) akışını bu ara bağlantılar olumlu yönde etkilemektedir. Ara bağlantılar matematik olarak özniteliklerinin boyutları eşit olan katmanlar arasında özniteliklerin toplama veya birleştirme (concatenate) işlemi sonucunda gerçekleşmektedir.

#### 4.1.2 Çok bağlantılı ağ

Klasik mimarilerde her bir katmanın girdisi önceki katmanın çıktısından oluşmaktayken çoklu bağlantılı ağlarda önceki katmanların çıktıları sıradaki katman tarafından kullanılabilir (Huang vd., 2017). Bu mimaride giriş katmanından çıkış katmanına doğru ilerleyince önceki katmanlarda çıkarılan öznitelikler sonraki katmanlarda



Şekil 4.3 Kodlayıcı-kod Çözücü evrişimli sinir ağlarının şematik çizimi



Şekil 4.4 Beş katmanlı çok bağlantılı derin model örneği (Huang vd., 2017)

da kullanılarak daha detaylı bilgilerin çıkarılmasına imkan sağlanmaktadır. Ayrıca çıkış katmanından girdi katmanına doğru geri iletilen kestirim hatası veya optimizasyon türevi sırayla birer birer katmanlardan geçirilmektense bağlantı noktalarında aynı zamanda birden fazla katmana aktarılmış olmaktadır. Böylece türevin geriye doğru yayılmasında sönümlenme sorunu, özellikle katman sayısı büyük olan modellerde, azaltılmış olmaktadır.

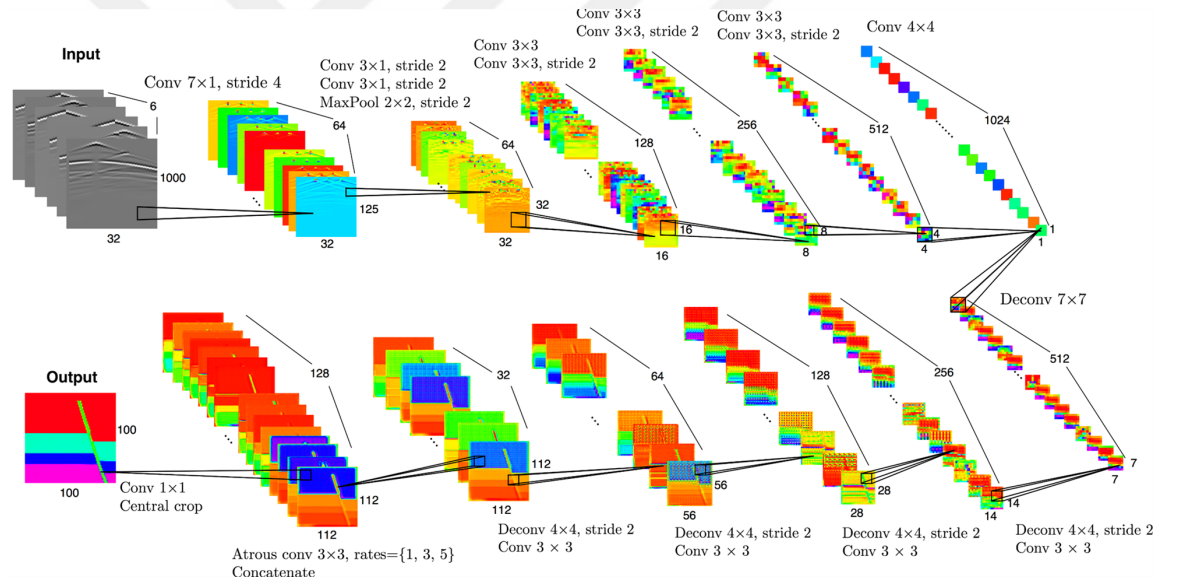
Şekil 4.4'te Huang vd. (2017) tarafından yayınlanan 5 katmanlı çok bağlantılı bir sinir ağı yapısı sunulmaktadır. Her katmanın çıktısı önceki katmanın/ katmanların çıktıları ile birleştirilerek sıradaki katmana aktarıldığı için çoklu bağlantılar öznelikleri aynı boyutta olan katmanların arasında kurulabilmektedir. Ard arda dizilmiş olan bu katmanlar çok bağlantılı veya yoğun blok (dense block) olarak tanımlanmaktadır. Örneğin Şekil 4.4'te görünen ağı beş katmanlı bir bloktur.

Yoğun blokların çıktısının farklı boyutta olan diğer bir bloka/katmana iletilebilmesi için boyut ayarlanması uygulayan *geçiş katmanından* geçirilmesi gerekmektedir. Geçiş katmanlarında öznelıkların sadece boyutu değil belki sayıları (kanal sayısı) da değiştirilerek derin modelin parametre sayısı da kontrol edilebilmektedir. Böylece çok bağlantılı bir derin model geçiş katmanlarla bir birine bağlanmış olan birden

fazla yoğun bloklardan oluşabilmektedir.

## 4.2 InversionNet Derin Modelin Mimarisi

Wu vd. (2018) tarafından *InversionNet* adı ile yayınlanan derin model kodlayıcı-kod çözücü şeklinde tasarlanmış olan bir evrişimli sinir ağıdır (Şekil 4.5). InversionNet sismik dalga şekli ile yeraltı hız modelin arasındaki regresyon ilişkisini öğrenerek uçtan-uca hız modellerini kestirebilmektedir. Bu çalışmada eğitim veri seti '*sismik atış kesitleri-hız modeli*' şeklinde hazırlanmıştır. Bu tez çalışması kapsamında temel model (baseline) olarak kullanıldığı için InversionNet derin modelin mimarisi ile ilgili gerekli bilgiler bu alt bölümde yer almaktadır.



Şekil 4.5 InversionNet derin modelin mimarisi (Wu vd., 2018)

Toplam 23 katmandan oluşan InversionNet derin modelin kodlayıcı birimi ilk 10 katmanı kapsamaktadır. Kodlayıcı birimin ilk üç katmanı  $1B$  evrişimli, dördüncü katman  $2 \times 2$  boyutlu maksimum havuzlama ve geri kalan yedi katmanı ise  $2B$  evrişimli çekirdeklerden oluşmaktadır. Aktivasyon fonksiyonu olarak ReLU fonksiyonu ve normalize etmek için de Bach normalizasyonu kullanılmıştır. Aslında her bir evrişimli katman evrişim çekirdeği (*Conv*), Bach normalizasyon (*BN*) ve ReLU aktivasyon fonksiyonu (*ReLU*) bileşenlerinden oluşmaktadır. Denklem 4.3'te bahse-

dilen evrişimli katmanların matematik ifadesi sunulmaktadır. Bu denklemde  $x^{(l)}$  ve  $x^{(l+1)}$  evrişimli katmanın sırayla girdi ve çıktı verilerini simgelemektedir.

$$x^{(l+1)} = ReLU(BN(Conv(x^{(l)}))) \quad (4.3)$$

Kodlayıcı birimin son katmanındaki öznitelikler  $4 \times 4$  boyutlu CNN çekirdeklerinden oluşan bir evrişimli katmandan geçirilerek darboğaz katmanının öznitelikleri  $1 \times 1 \times 1024$  boyutunda oluşturulmaktadır. Bu katmanda oluşan öznitelikler derin modelin ikinci birimi yani kod çözücü birimine aktarılıp özniteliklerin boyutu git-tikçe büyütülmektedir.

12 katmandan oluşan kod çözücü biriminde özniteliklerin üst örnekleme (upsampling) işlemi dekonvolüsyon çekirdekleri ile yapılmaktadır. Darboğaz katmanından gelen öznitelikler  $7 \times 7$  boyutlu dekonvolüsyon çekirdekleri ile büyütülerek diğer katmanlara aktarılmaktadır. Modelin devamında dekonvolüsyon ve evrişimli katmanlar ard arda dizilerek sıradaki sekiz katmanı oluşturmaktadır. Yirminci katmanın çık-tısı genişleme oranları 1, 3 ve 5 olan üç genişletilmiş katmana iletilerek aynı boyutta öznitelikler üretilmektedir. Eşit boyutta çıkarılan öznitelikler birleştirilerek son katmana iletilmektedir.  $1 \times 1$  boyutlu evrişimli çekirdeklerin ürettiği nihai öznitelik kırılarak derin modelin hız haritası kestirimi olarak sunulmaktadır.

Bu tez çalışmasında InversionNet derin modelin tasarımında bazı değişiklikler uygulanarak temel model olarak kullanılmıştır. Li vd. (2020) de yayınladıkları çalışmada temel model olarak InversionNet'i kullanmışlardır. Ancak onlar da bazı değişiklikler uygulayarak ilk sürümüne göre daha iyi sonuçlar almışlardır. Kolaylık maksatı ile InversionNet'in ilk sürümü (Wu vd., 2018), ikinci sürümü (Li vd., 2020) ve bu tez çalışmasında sunulan üçüncü sürümü sırasıyla  $\alpha$ ,  $\beta$ , ve  $\gamma$  olarak belirtilmiştir. Tezin bir sonraki bölümünde bu üç sürümün kestirim hata miktarı ve benzerlik değerleri sunulmuştur. InversinNet'in  $\beta$  ve  $\gamma$  versiyonları izleyen şekilde anlatılmaktadır.

- **InversionNet ( $\beta$ ):** Li vd. (2020) InversionNet derin modelin dördüncü katmanında yer alan maksimum havuzlama katmanının yerine  $1B$  evrişimli katman

kullanılmışlardır. Ayrıca kod katmanındaki özniteliklerin boyutu  $1 \times 1$  yerine  $3 \times 3$  boyutundadır. Diğer bir deyişle özniteliklerin boyutu  $3 \times 3$ 'ün altına inmemektedir. Her iki sürümün parametre sayısı yaklaşık 40 milyondur.

- **InversionNet ( $\gamma$ )**: Bu tez çalışması kapsamında InversionNet'in kod çözücü birimi değiştirilerek yeni sürümü elde edilmiştir. Önceki sürümlerin ikisinde de kod çözücü birimindeki üst örnekleme işlemi dekonvolüsyon çekirdekleri aracılığı ile yapılmaktayken  $\gamma$  sürümünde *üst örnekleme* fonksiyonu kullanılmaktadır. Bununla birlikte  $\beta$  versiyonunda olduğu gibi bu versiyonda da dördüncü katman  $1B$  evrişimli katmanla değiştirilmekte ve kod katmanının öznitelik boyutu  $3 \times 3$  olarak ayarlanmaktadır. Bahsedilen değişikliklerin sonucunda elde edilen yeni versiyonun aynı girdi verileri için parametre sayısı 40 miyondan 20 milyona düşürülmüş olmaktadır.

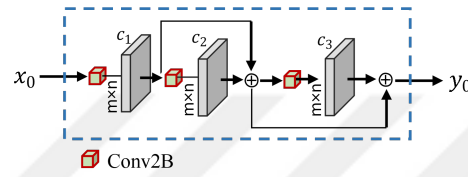
### 4.3 SVInvNet Derin Modelin Tasarımı

SVInvNet derin model yoğun bloklardan oluşan çok bağlantılı kodlayıcı - kod çözücü şeklinde tasarlanmıştır. Tasarlanan mimaride bir taraftan girdi verilerinden çıkarılan öznitelikler hız modelin uzamına ve boyutuna aktarılmakta diğer taraftan ise bu aktarma işleminde girdi ve çıktı verilerin arasında ileriye ve geriye yönelik bilgi akışı kolaylaştırılmıştır. SVInvNet olarak elde edilen son tasarım (Şekil 4.7) kademeli şekilde yapılmış olan denemelerin sonucu dikkate alınarak ortaya çıkmıştır.

InversionNet derin modelde olduğu gibi SVInvNet'in ilk dört katmanı  $1B$  evrişimli çekirdeklerden oluşmaktadır. Dördüncü katmandan çıkan özniteliklerin düşey eksen yatay boyutla eşit olacak kadar küçülmektedir. Diğer bir deyişle zaman eksenindeki örnek sayısı alıcı sayısı ile eşit olmaktadır. Üçüncü katmandan sonra dördüncü katmanı da içerecek şekilde yoğun bloklar konularak mimarinin kodlayıcı ve kod çözücü birimleri oluşturulmaktadır. Bu nedenle de mimarinin birimlerine geçmeden önce SVInvNet'te kullanılan yoğun blokların yapısı anlatılmaktadır.

### 4.3.1 Yoğun blokların yapısı

Her bir yoğun blok imge boyutları sabit olan üç adet  $2B$  evrişim katmanından oluşmaktadır. Bloka girdi olarak iletilen imge boyutu değişmeden çıktı olarak diğer bloka veya geçiş katmanına iletilmektedir. Yoğun blokun içindeki katmanların girdisi önceki katmanların birleşmiş çıkılarından oluşmaktadır. Yoğun blokun çıktı imgesi üç katmanın çıktı imgelerinin birleşmesinden oluşmaktadır (Şekil 4.6).



Şekil 4.6 Üç katmandan oluşmuş yoğun blok tasarımı. Katmanlardaki imge boyutu sabittir  $m \times n$

Şekil 4.6'da üç katmandan oluşan SVInvNet derin modelin tasarımında kullanılan yoğun blok yapısı gösterilmektedir. Tasarlanan yoğun blokların matematik ifadesi Denklem 4.4 - 4.8'de sunulmaktadır.  $Conv2B$  olarak şekilde çizilen küpler  $3 \times 3$  boyutlu evrişim çekirdeği, bach normalizasyonu (BN) ve ReLU aktivasyon fonksiyonundan oluşmaktadır ( $Conv2B = conv2B + BN + ReLU$ ). Kolaylık amacı ile tezin ilerleyen bölümlerinde de  $Conv2B$  aynı anlamda kullanılmaktadır.

$$x_1^i = L_1^i(x_0) ; C_{out} = c_1 \quad (4.4)$$

$$x_2^i = L_2^i(x_1) ; C_{out} = c_2 \quad (4.5)$$

$$x_{1,2}^i = concatenate(x_1^i, x_2^i) \quad (4.6)$$

$$x_3^i = L_3^i(x_{1,2}^i) ; C_{out} = c_3 \quad (4.7)$$

$$y_0 = \text{concatenate}(x_1^i, x_2^i, x_3^i) \quad (4.8)$$

Burada  $L_j^i$ , i. yoğun blokun j. katmanı,  $x_0$  ve  $y_0$  sırasıyla yoğun blokun girdisi ve çıktısı ve  $x_j^i$  ise j. katmanın çıktısıdır.  $c_j$  katmanların çıktı boyutunun derinliğini (kanal sayısı) belirleyen imge veya öznetelik sayısını (CNN filtreleri) göstermektedir. SVInvNet derin modelin mimarisinde  $c_j = 64$ , ve yoğun blokun çıktı verisinin ( $y_0$ ) kanal sayısı ise 192 ve  $L_j$   $2B$  evrişimli katmandır ( $Conv2B$ ). Şekil 4.7’de çizilmiş olan SVInvNet derin modelin mimarisinde yoğun bloklar mavi renkli dikdörtgen prizmalarla simgelenmektedir.

Yoğun blokların tasarımından anlaşıldığı gibi önceki katmanların öznetelikleri birleşerek sıradaki katmana aktarılmaktadır. Özneteliklerinin boyutu eşit olan yoğun blokların çıktıları da birleşerek diğer yoğun bloklara veya katmanlara aktarılmaktadır. Ancak birleştirme işlemi giderek kanal sayısını çoğaltmakta ve derin modelin parametre sayısı ve hesaplama yükünü artmaktadır. Bu sorunun giderilmesi amacıyla yoğun blokların çıkışlarında veya gerekli yerlerde geçiş katmanları kullanılmaktadır. Geçiş katmanları  $1 \times 1$  ve  $3 \times 3$  boyutlu evrişimli çekirdekler içeren evrişimli katmandan oluşmaktadır. Bu katmanlar verilerin kanal sayısını azaltarak ve/veya boyutunu küçülterek çok bağlantılı bloklar ve katmanların kullanılmasına imkan sağlamaktadır.

### 4.3.2 Kodlayıcı ağı özelliği

Tasarlanan mimarinin ilk dört katmanında  $1B$  CNN’ler (Şekil 4.7’de turuncu oklar ile gösterilmiştir) kullanılarak girdi verilerindeki zaman boyutu alıcıların sayısı ile eşlenecek kadar küçültülmektedir. Diğer bir deyişle adım boyu ve dolgu değişkenleri ayarlanarak dördüncü katmanda  $34 \times 34$  boyunda olan imgeler üretilmektedir. Kodlayıcı birimin ilerleyen bölümü boyutları giderek küçülen yoğun bloklardan oluşmaktadır.  $2B$  CNN’ler kullanılarak imgelerin boyutu  $18 \times 18$ ,  $9 \times 9$  ve  $6 \times 6$  olarak küçültülmektedir. SVInvNet derin modelin mimarisinde  $34 \times 34$  boyutunda olan im-

gelere 3 yoğun blok ve diğer boyutlar için de sırayla 2, 1 ve 1 yoğun blok ayrılmıştır.

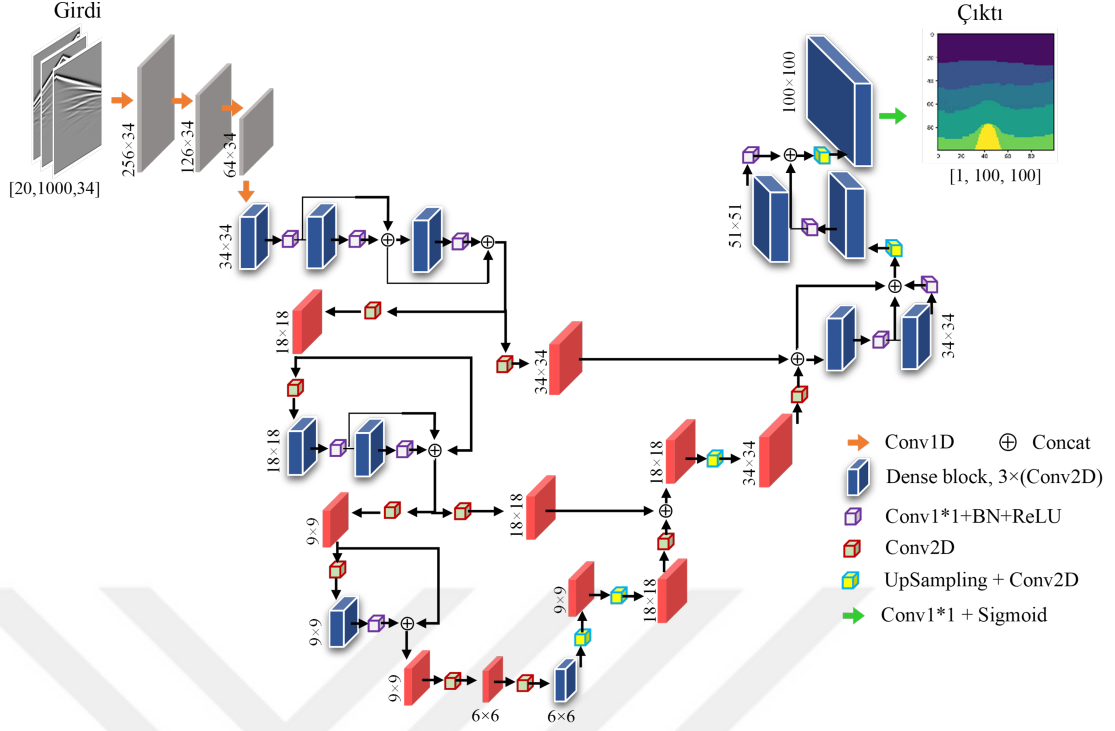
$34 \times 34$  boyutunda olan ilk yoğun blok dördüncü katmanı içererek şekilde üçüncü katmandan sonra yer almakta ve onun ardınca diğer iki  $34 \times 34$ 'lük yoğun bloklar da sıralanmaktadır. Yoğun blokların çıktıları da çok bağlantılı mantık esasında birleştirilip  $1B$  evrişimli geçiş katmanından geçirilerek diğer bloklara aktarılmaktadır. Şekil 4.7'de bu geçiş katmanları mor-beyaz renkli küçük küpler ile gösterilmektedir. Her üç yoğun bloğun çıktısı birleşerek  $2B$  evrişimli bir geçiş katmanından geçilerek  $18 \times 18$  boyutuna küçültülmektedir. Bu katmanın çıktısı da  $2B$  evrişimli bir geçiş katmanından geçilerek aynı boyutta olan yoğun bloka aktarılmakta bu bloğun çıktısı da sıradaki yoğun bloka iletilmektedir.  $2B$  evrişimli geçiş katmanları Şekil 4.7'de kırmızı-yeşil renkli küçük küplerle gösterilmektedir.

Aynı yol izlenerek  $18 \times 18$ 'lik öznitelikler  $9 \times 9$  boyutuna indirgenip tek bir yoğun bloka aktarılmaktadır. Yoğun bloğun çıktısı bloka girdi olarak sunulan öznitelikler ile birleştirilerek geçiş katmanından geçirilip kodlayıcı birimin  $9 \times 9$  boyutunda olan son öznitelikleri çıkarılmaktadır. Elde edilen bu öznitelikler  $2B$  evrişimli katmanından geçirilerek  $6 \times 6$  boyutunda olan kod katmanının imgeleri oluşmaktadır.

### 4.3.3 Kod çözücü ağın özellikleri

Kod veya darboğaz katmanındaki öznitelikler  $2B$  evrişimli geçiş katmanından geçirilerek kod çözücü birimindeki  $6 \times 6$  boyutlu yoğun bloka aktarılmaktadır. Kod çözücü biriminde imgelerin büyütülme işlemi üst örnekleme fonksiyonu ile yapılmaktadır. Üst örnekleme fonksiyonu  $2B$  evrişimli bir katmanla birleşerek özniteliklerin boyutu büyültmektedir. *Üst örnekleme katmanı* olarak adlandırılan bu birleşik işlem Şekil 4.7'de sarı-yeşil renkli küçük küplerle gösterilmektedir.

İlk yoğun bloğun çıktısı  $6 \times 6$  boyutlu imgeler iki üst örnekleme katmanından geçirilerek sırasıyla  $9 \times 9$  ve  $18 \times 18$  boyutuna dek büyütülmektedir.  $2B$  evrişimli geçiş katmanından geçirilen  $18 \times 18$  boyutlu imgeler kodlayıcı birimindeki eşit boyutlu



Şekil 4.7 Sismik hız modeli ters çözümü için tasarlanan SVInvNet derin model mimarisi

özniteliklerle birleşerek sıradaki üst örnekleme katmanına sunulmaktadır.  $34 \times 34$  boyutunda olan bu katmanın çıktısı  $2B$  evrişimli geçiş katmanından geçirildikten sonra kodlayıcı katmanından alınan  $34 \times 34$  boyutlu özniteliklerle birleşerek aynı boyutta olan bir yoğun bloka aktarılmaktadır. Ard arda dizilen  $34 \times 34$  boyutlu iki yoğun blokun çıktıları aynı boyutta olan girdi imgeleri ile birleşerek üst örnekleme katmanına iletilmektedir.  $51 \times 51$  boyutunda olan çıktı öznitelikleri aynı boyutta olan yoğun bloka iletilmektedir.

Ard arda dizilmiş olan  $51 \times 51$  boyutlu yoğun blokların çıktısı birleştirilerek üst örnekleme katmanından geçirilip hız modellerin boyutu,  $100 \times 100$ , ile eşdeğer olacak şekilde büyütülmektedir.  $100 \times 100$  boyutlu öznitelikler aynı boyutta olan son yoğun bloka sunulmaktadır. Bu yoğun blokun çıktısı  $1 \times 1$  boyutlu evrişimli çekirdeklerden oluşan ve aktivasyon fonksiyonu Sigmoid olan kod çözücü birimin son katmanından geçirilerek derin modelin hız modeli kestirimi veya çıktısı üretilmiş olmaktadır.

Özet olarak anlatılırsa kod çözücü biriminde özniteliklerin boyutu üst örnekleme

katmanlarında  $9 \times 9$ ,  $18 \times 18$ ,  $34 \times 34$ ,  $51 \times 51$  ve  $100 \times 100$  olarak kademeli şekilde büyütülmektedir. Tasarlanan derin modelin genelleme yeteneğini güçlendirme amacıyla  $34 \times 34$  ve  $51 \times 51$  boyutlu imgeler için iki ve  $100 \times 100$  boyutlu imgeler için de bir yoğun blok ayrılmıştır. Bilgi akışının kolaylaştırılması ve kod çözücü ağının konsolide edilmesi amacıyla kodlayıcı biriminden aktarılan imgeler ( $18 \times 18$  ve  $34 \times 34$  boyutlu imgeler)  $2B$  evrişimli geçiş katmanından geçirilmektedir.

Yoğun bloklar ve çok bağlantılı topolojisi olan SVInvNet'in tasarımında derin modelin parametre sayısı geçiş katmanları ile kontrol edilerek öznitelikler sayı olarak çoğaltılabilir ve boyutu olarak da büyütülebilir durumdadır. Son katmandan önce kullanılan  $100 \times 100$  boyutlu yoğun bloğun kullanılması derin modelin eğitim ve test aşamasındaki kestirim hata payını azaltarak performansını etkilemektedir.

Girdi verilerin boyutu,  $[20, 1000, 34]$ , dikkate alınarak SVInvNet derin modelin parametre sayısı yaklaşık 4 milyona ulaşmaktadır. SVInvNet'in tüm parametreleri eğitilebilir olmakta ve eğitim sürecinde belirlenmektedir. Derin modelin kestirimi amaç verileri, gerçek hız modelleri, ile eşit boyutta,  $[1, 100, 100]$ , olmaktadır.

## 5. EĞİTİM AŞAMASI

Üçüncü bölümde anlatıldığı gibi gürültüsüz ve gürültülü olarak ikiye ayrılmış olan eğitim veri kümesi sayı olarak beş alt kümeden, TD-I, TD-II, TD-III, TD-IV, ve TD-V, oluşmaktadır. Anlatımda kolaylık olsun diye eğitilmiş derin modeller eğitim veri kümesinin Roma sayısı ile ifade edilerek sırası ile Model-I, Model-II, Model-III, Model-IV, ve Model-V şeklinde yazılmaktadır. Test denektaşı, gürültüsüz ve gürültülü verilere özgü olarak tüm eğitilmiş modeller için sabit bir veri kümesinden oluşmaktadır.

Derin modelin hiperparametreleri tüm eğitim süreçleri için farklı denemelerin sonucunda belirlenmiştir. Optimizasyon için Adam algoritması (Kingma ve Ba, 2014) kullanılmıştır. Eğitim süresinde doğrulama (validation) verilerinin hata değerlerine göre küçültülen eğitim adım boyunun başlangıç değeri  $5e - 3$  olarak ayarlanmıştır. Eğitim süresi tüm eğitim senaryolarında 500 epok olarak sabitlenmiş ve bu sürede eğitilmiş derin model olarak hesaplanmış katsayı veri kümesi doğrulama yanılğı enerjisine göre kayıt edilmiştir.

Eğitim aşamasında derin modelin kestirimleri ile gerçek hız modellerin arasındaki uyumsuzluk hatasının ve benzerlik ölçütünün hesaplanması için sırayla *ortalama mutlak hata* (mean absolut error - L1) ve *yapısal benzerlik indeks ölçüsü* (structural similarity index measure - SSIM) kullanılmıştır. L1 ve SSIM fonksiyonları ayrı ayrı katsayılarla çarpılarak yanılğı enerjisi daha etkin hale getirilmiştir. L1 fonksiyonu için  $\lambda_{L_1}$  olarak tanımlanan bu katsayı tüm eğitim süresince 40 olarak sabitlenmiştir. SSIM fonksiyonunun katsayısı ise  $\lambda_{SSIM}$  olarak tanımlanmış ve kırkıncı epoka dek 160 ve ondan sonraki epoklar için 250 olarak belirlenmiştir. Diğer bir deyişle yanılğı enerjisi  $\lambda_{L_1}L1 + \lambda_{SSIM}(1 - SSIM)$  şeklinde kullanılarak optimizasyon sürecinde minimuma indirgenmesi hedeflenmiştir.

Test aşamasında ise hata ve benzerlik değerlerini ölçmek için L1 ve SSIM ölçütlerinin

yanısına *ortalama hata kareleri* (mean squared error - L2) ve *çoklu yapısal benzerlik indeks ölçüsü* (multi-structural similarity index measure - MSSIM) de kullanılmıştır. Vurgulamak gerekir ki hata ve benzerlik değerleri normalize edilmiş tüm test veri kümesi üzerinde hesaplanıp ortalama miktarı elde edilmiştir.

Eğitim veri kümeleri derin modele sunulmadan önce normalizasyon işlemi yapılmaktadır. Böylece sinir ağları verilerin dağılımını kolayca kestirebilmektedir. Bu nedenle de hız modelleri en büyük hız değerine,  $4550m/s$ , bölünerek  $0 - 1$  aralığına çekilmiştir. Sismik atış kesitleri de standart dağılımı kullanılarak normalize edilmiştir. PyTorch kütüphanesi kullanılarak tüm eğitim ve test süreçleri NVIDIA RTX A4500 grafik kartlı bilgisayarda GPU çekirdeklerinde çalıştırılmıştır.

### 5.1 InversionNet ( $\alpha$ ), ( $\beta$ ) ve ( $\gamma$ )

Gürültüsüz ve gürültülü veri kümelerinden TD-V eğitim veri seti kullanılarak InversionNet Derin modelin her üç versiyonu, ( $\alpha$ ), ( $\beta$ ) ve ( $\gamma$ ), eğitilmiştir. Performans değeri olarak diğer versiyonlardan daha iyi sonuçlar sunan ( $\gamma$ ) versiyonu temel model olarak seçilmiştir. Buradaki amaç üç farklı versiyonun karşılaştırması olduğu için eğitim aşaması sadece TD-V eğitim veri kümesi ile gerçekleşmiştir. Test için de gürültüsüz ve gürültülü veri kümelerinin test denektaşı kullanılmıştır.

InversionNet derin modelin her üç versiyonunun test aşamasındaki performansı, gürültüsüz ve gürültülü veri kümeleri için Çizelge 5.1'de sunulmaktadır. Elde edilen hata ve benzerlik değerleri dikkate alınarak gürültüsüz veri kümesinde InversionNet( $\gamma$ ) versiyonu düşük hata, L1 ve L2, ve yüksek benzerlik, SSIM, değerleri ile hız modellerini kestirebilmiştir. Gürültülü veri kümesi için InversionNet'in ( $\gamma$ ) ve ( $\beta$ ) versiyonlarının ürettiği hata ve benzerlik değerleri birbirine çok yakın olmaktadır.

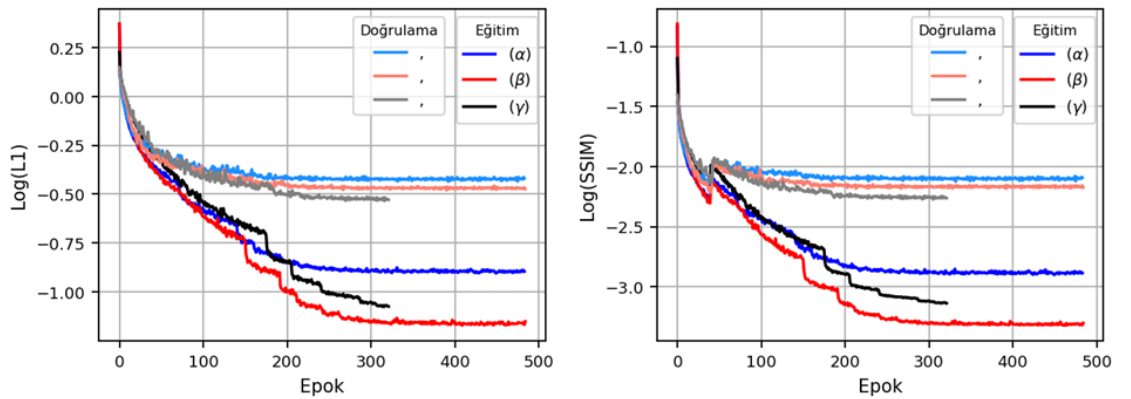
Üç versiyonun eğitim esnasındaki eğitim ve doğrulama yanığı enerji eğrileri gürültüsüz veriler için Şekil 5.1'de sunulmaktadır. L1 ve SSIM yanığı enerjileri ayrı ayrı çizilip eğriler kolayca ayırt edilsinler diye düşey eksen logaritmik olarak ölçek-

Çizelge 5.1 InversionNet  $\alpha$ ,  $\beta$  ve  $\gamma$  versiyonlarının gürültüsüz ve gürültülü test verileri üzerindeki performansı. Gürültüsüz ve gürültülü olarak TD-V eğitim veri kümesi kullanılmıştır

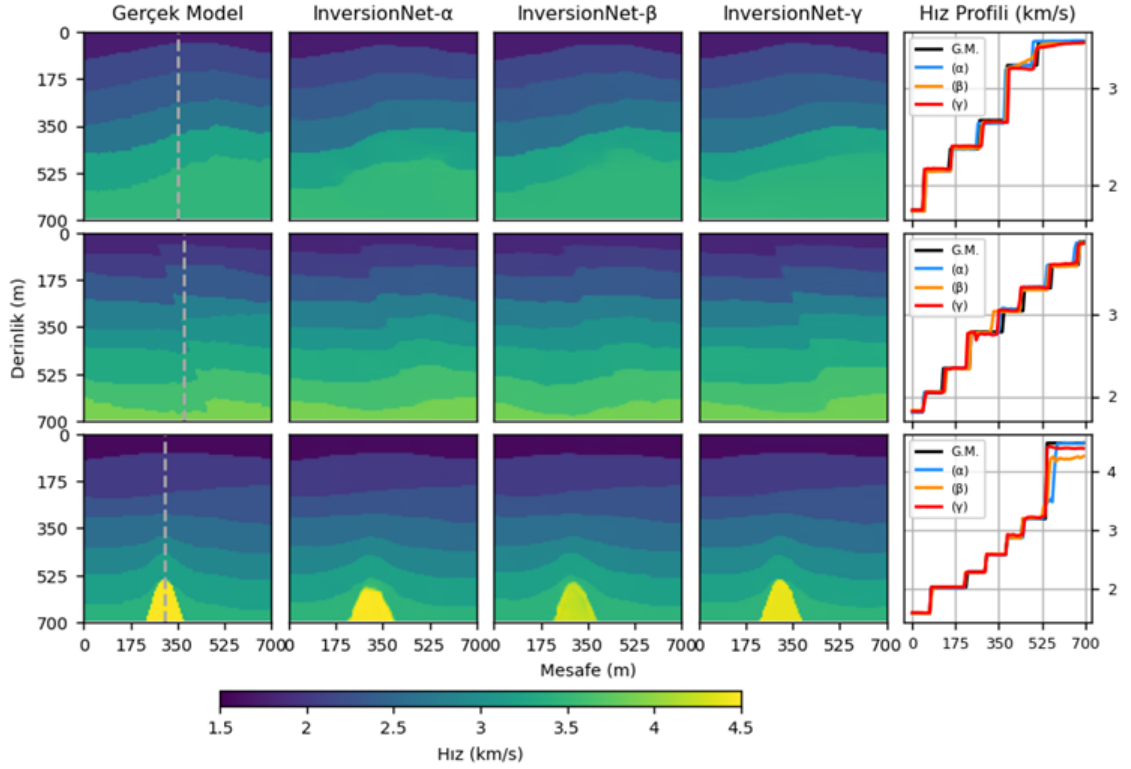
InversionNet	Gürültüsüz Veri Kümesi			Gürültülü Veri Kümesi		
	( $\alpha$ )	( $\beta$ )	( $\gamma$ )	( $\alpha$ )	( $\beta$ )	( $\gamma$ )
L1	0.008828	0.008385	0.007316	0.014378	0.013028	0.013142
L2	0.000499	0.000471	0.000401	0.000765	0.000706	0.000713
SSIM	0.999970	0.999972	0.999978	0.999943	0.999950	0.999949
MSSIM	0.999997	0.999997	0.999997	0.999994	0.999995	0.999995

lenmiştir. Mavi, siyah ve kırmızı renkli olan eğriler eğitim ve onların soluk versiyonu doğrulama verilerine aittir. Çizilen eğrilerden anlaşıldığı gibi epok sayısı artınca hata payı azalmakta ama bu düşüş trendi belli bir epoktan sonra durağan hale gelmiştir. ( $\beta$ ) versiyonunun eğitim yanlgı enerjileri, L1 ve SSIM, diğer iki versiyona kıyasla daha düşük değerlere ulaşmaktadır (Kırmızı eğriler). Ancak doğrulama yanlgı enerjilerinin eğrileri aynı yönde hareket etmemekte ve ( $\gamma$ ) versiyonuna ait eğriler, L1 ve SSIM, en alt kısımda yer almaktadır (gri renkli eğriler). Diğer bir deyişle genelleme potansiyeli olarak ( $\gamma$ ) versiyonu en yüksek performansı göstermektedir.

Eğitilmiş derin modellerin performansının nitel olarak da karşılaştırılması amacı ile test veri kümesinden rastgele örnekler Şekil 5.2'de gerçek hız modeli ile birlikte yan yana gösterilmektedir. Örnekler tabakalı, faylı ve tuz domu içeren hız modellerine



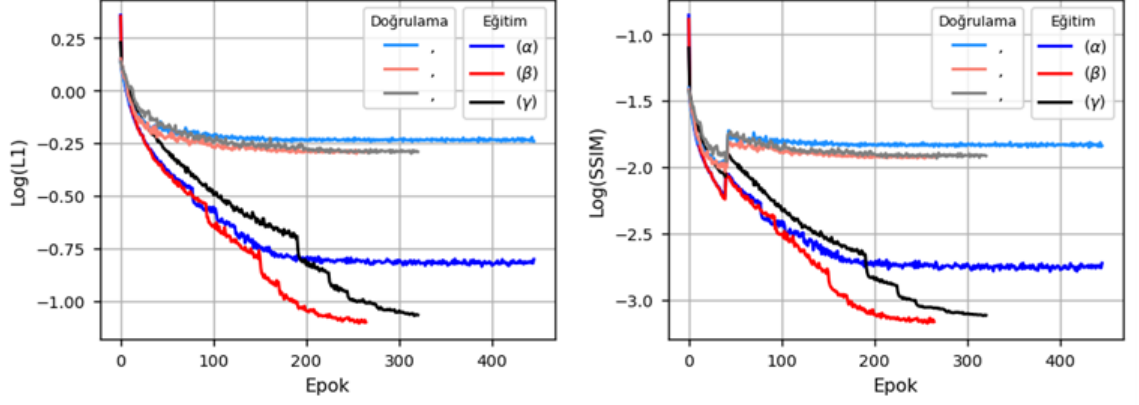
Şekil 5.1 InversionNet derin modelin ( $\alpha$ ), ( $\beta$ ) ve ( $\gamma$ ) versiyonlarına ait eğitim ve doğrulama yanlgı enerjileri. Gürültüsüz TD-V eğitim veri kümesi kullanılmıştır



Şekil 5.2 InversionNet derin modelin üç farklı versiyonunun kestirdiği hız modelleri ve gerçek hız modeli. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen hız profili yer almaktadır. Modeller gürültüsüz TD-V eğitim veri kümesi ile eğitilmiştir

aittir. Şeklin son sütununda gerçek hız modeli üzerinde gri renkli kesik çizgi ile gösterilen hattın hız profilleri çizilmektedir. İlk satırda yer alan tabakalı hız modelinde üç farklı versiyonun performans farklı beşinci ara yüzeyde bariz bir şekilde ayırt edilmektedir.

İkinci satırda yer alan faylı hız modelin üç farklı kestirimi bir biri ile karşılaştırılınca fay hattının alt tabakalardaki devamlılığı ve atım boyu açısından  $InversionNet(\gamma)$ 'nin ürettiği sonuç daha çok gerçek modeli yansıtmaktadır. Tuz domu içeren modelde tuz domu altıncı ara yüzeyden geçerek altıncı tabakaya girmiş durumdadır. Ancak  $InversionNet(\alpha)$  ve  $(\beta)$  tuz domunu yedinci tabakanın içinde sırasıyla daha geniş ve daha düşük hızla kestirmiştir. Ama  $(\gamma)$  versiyonu oldukça gerçek modele yakın bir hız modeli sunmaktadır.

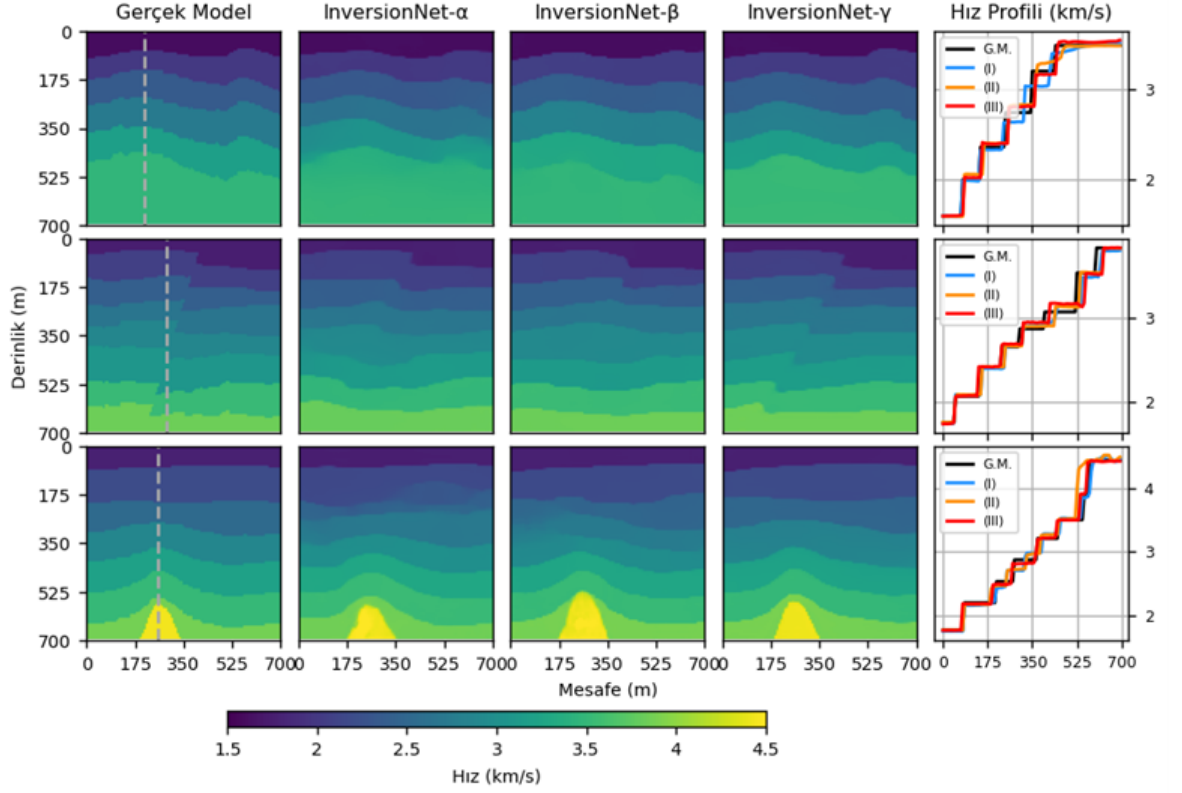


Şekil 5.3 InversionNet derin modelin üç versiyonuna ait eğitim ve doğrulama yanılı enerji. Gürültülü TD-V eğitim veri kümesi kullanılmıştır

Gürültülü veri kümesinin yanılı enerji eğrileri  $(\alpha)$ ,  $(\beta)$  ve  $(\gamma)$  versiyonları için Şekil 5.3'te gösterilmektedir. Bu çizgilerde de en düşük eğitim yanılı enerjisi  $(\beta)$  versiyonuna aittir ama doğrulama eğrilerinde  $(\beta)$  ve  $(\gamma)$  versiyonuna ait eğriler neredeyse aynı değerleri almaktadır. Doğrulama yanılı enerjileri eğitilmiş modelin genelleme potansiyelini göstermektedir. Eğitim yanılı enerjisi düşük olup doğrulama aşamasında düşük hatalara ulaşamayan modeller eğitim veri kümesini ezberlemektedir.

Gürültülü test denektaşına ait üç hız modeli ve eğitilmiş InversionNet $(\alpha)$ ,  $(\beta)$  ve  $(\gamma)$  versiyonlarına ait ilgili kestirimler Şekil 5.4'te gösterilmektedir. En son sütunda da gerçek hız modelinde gri renkli kesik çizgi ile işaretlenen kısmın hız profili çizilmektedir. Tabakalı hız modelinde ufak farklarla tekrarlanan dalgalı ara yüzeyler özellikle beşinci tabakanın şekli üç farklı versiyonun zorluk yaşadığı kısım olmaktadır. Ancak gerçek modele en yakın model Inversionnet $(\gamma)$  tarafından sunulmaktadır. Bu durum faylı model için de geçerlidir.

Şekil 5.4'teki örneklerde gürültülerin etkisi tuz domu modelinde daha belirgin bir şekilde görünmektedir. İkinci ve üçüncü ara yüzeylerin şekli  $(\alpha)$  ve  $(\beta)$  versiyonların sunduğu sonuçlarda bulanık ve bir birine girmiş şekilde kestirilmektedir. Tuz domunun gövde şekli ve uç kısmının altıncı ara yüzeye olan mesafesi de  $(\alpha)$  ve  $(\beta)$  versiyonlarında doğru bir şekilde kestirilmemiştir. InversionNet $(\gamma)$  tarafından kestirilen hız modelinde bu sorunlar görünmemektedir.



Şekil 5.4 Gerçek hız modeli ve InversionNet derin modelin üç farklı versiyonunun sonuçları. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen hız profili yer almaktadır. Modeller gürültülü TD-V eğitim veri kümesi ile eğitilmiştir

InversionNet derin modelin her üç versiyonunun gürültüsüz ve gürültülü veri kümelerindeki performans değerleri, eğitim ve doğrulama yanlış enerjilerinin akımı ve nitel karşılaştırılması dikkate alınarak InversionNet( $\gamma$ ) versiyonu temel model olarak kabul edilmiştir.

## 5.2 SVInvNet

SVInvNet derin modelin eğitimi beş alt veri kümesi, TD-I, TD-II, TD-III, TD-IV, ve TD-V, kullanılarak ayrı ayrı gerçekleştirilmiştir. Gürültüsüz ve gürültülü verilerle eğitilmiş beş model ayrıca ilgili test veri setleri ile test edilmiştir. Eğitilmiş derin modellerin test aşamasındaki hata ve benzerlik değerleri gürültüsüz veri kümesi için Çizelge 5.2'de sunulmaktadır. Elde edilen sonuçların temel model ile karşılaştırıl-

ması için InversionNet( $\gamma$ )-V modeline ait test aşamasındaki performans değerleri de çizelgenin son satırına eklenmiştir.

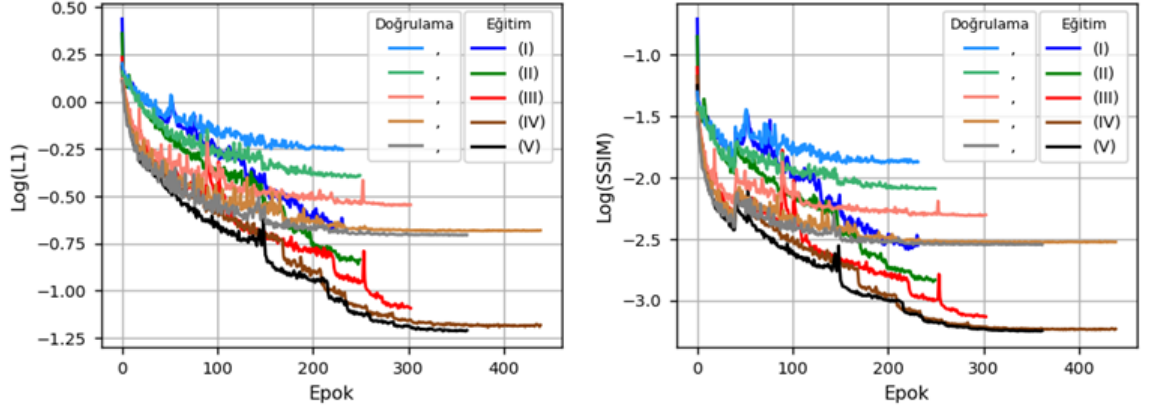
Çizelge 5.2’de L1 ve L2 hata değerleri eğitim veri kümesinin artması ile azalmakta ve benzerlik ölçütleri, SSIM ve MSSIM, büyümektedir. Böylece eğitim veri kümesi artınca derin modelin kestirimleri düşük hata ve yüksek benzerlik değerleri ile sonuçlanması anlaşılmaktadır. Çizelgedeki koyu renkle gösterilen en iyi değerler Model-V’e aittir. Temel model ile karşılaştırılınca TD-III ile eğitilen SVInvNet (Model-III) InversionNet( $\gamma$ )-V’e göre daha düşük hata değerleri ile hız modellerini kestirebilmesi açıkça anlaşılmaktadır.

Gürültüsüz verilerle eğitilen SVInvNet’in tüm beş eğitilmiş modelinin eğitim ve doğrulama yanılğı enerji eğrileri Şekil 5.5’te gösterilmektedir. Koyu renkler eğitim ve onların soluk versiyonu ise doğrulama verilerine ait olan L1 ve SSIM eğrileri ayrıca çizilmektedir. Epok sayısı artınca daha düşük değerlere varan yanılğı eğrilerin düşüş eğimi de epok sayısı ile küçülmektedir. Diğer bir deyişle ilk başlarda yanılğı enerjisi daha hızlı bir şekilde azalırken ilerleyen epoklarda düşüş eğimi küçülerek sıfıra yaklaşmıştır.

Şekil 5.5’te görüldüğü gibi L1 ve SSIM eğrilerinde, ister eğitim ister doğrulama verileri için, en az ve en çok düşüşler sırası ile Model-I ve Model-V’in yanılğı enerjilerine aittir. Diğer eğitilmiş modeller de bu iki eğrinin arasında yer almaktadır.

Çizelge 5.2 SVInvNet derin modelin test veri seti üzerindeki performansı. Model-I den Model-V dek gürültüsüz eğitim veri kümeleri, TD I-V, kuulanılarak elde edilmiştir. InversionNet( $\gamma$ )TD-V veri kümesi ile eğitilmiştir

	L1	L2	SSIM	MSSIM
Model-I	0.013680	0.000737	0.9999436	0.99999491
Model-II	0.009833	0.000509	0.9999674	0.99999692
Model-III	0.006844	0.000356	0.9999810	0.99999812
Model-IV	0.005301	0.000272	0.9999870	0.99999870
Model-V	<b>0.004944</b>	<b>0.000253</b>	<b>0.9999884</b>	<b>0.99999882</b>
InversionNet( $\gamma$ )-V	0.007316	0.000401	0.999978	0.999997



Şekil 5.5 SVInvNet derin modelin L1 ve SSIM yanlıgı enejilerine ait 5 farklı eğitim ve doğrulama eğrisi. Eğriler TD I-V gürültüsüz veriler ile eğitilen modellere aittir

Eğitim veri sayısının artması eğrilerin arasındaki farkı azalmakta öyle ki Model-IV ile Model-V'in eğrileri, özellikle SSLM yanlıgı eğrisinde, 200. epoktan sonra birbiri ile örtüşmektedir. Önceki bölümlerde, özellikle üçüncü ve dördüncü bölümlerde, anlatıldığı gibi sinir ağların hata payı sadece eğitim veri kümesinin büyütülmesi ile azalmamakta ve derin modelin mimarisi de etkili olmaktadır.

SVInvNet derin modelin nitel performansı, farklı eğitim veri sayısı için, test veri kümesinden rastgele seçilen hız modelleri ile Şekil 5.6-5.9 sunulmaktadır. Sırasıyla tabakalı, faylı ve tuz domu içeren veri gruplarına ait olan bu şekiller gürültüsüz veri kümesine aittir. İlk sütunda gerçek hız modeli ve son sütunda ise gerçek hız modelinde gri renkli kesik çizgi ile gösterilen şeritin hız profili çizilmektedir. Her üç şekilde de en iyi sonuçlar Model-V'e aittir.

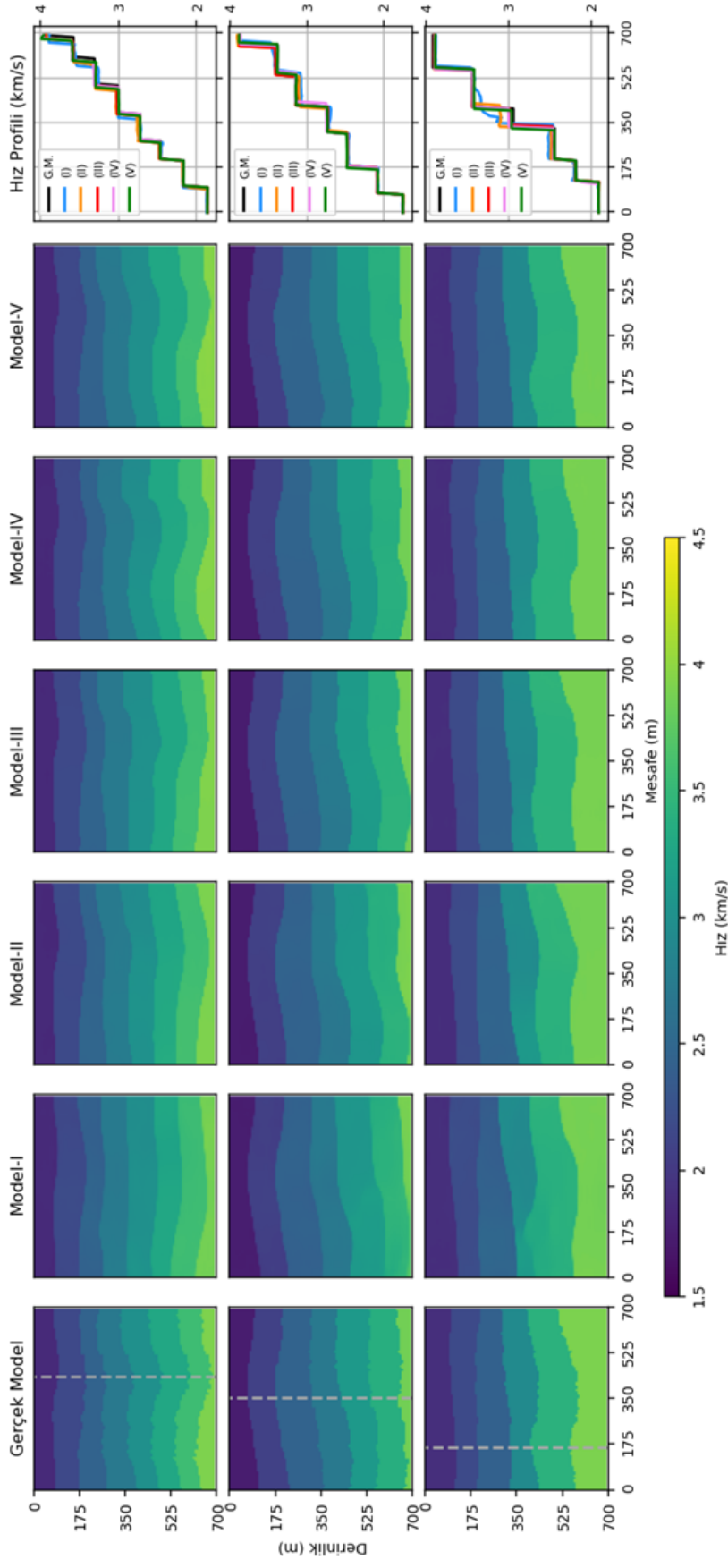
Tabakalı modellere ait Şekil 5.6'da ilk satırda ara yüzeylerin tekrarlanan dalgalı akımı, ikinci satırda en son ara yüzeyin oldukça derinde yer alması ve son satırda ise dördüncü tabakanın inceliği beş farklı modelin arasındaki yarışma konusunu oluşturmaktadır. Eğitim veri kümesinin artması gerçek modeldeki ara yüzeylerin genel şeklini yansıtan kestirimlerden detayları da içeren kestirimlerin ortaya çıkmasına neden olmaktadır. Model-V'in ürettiği sonuçlar daha derin seviyelerde yer alan ara yüzeylerin veya ince tabakaların şeklini daha düşük hata payı ile kestirmektedir.

Şekil 5.7’de sunulan derin model çıktıları dikkate alınarak, neredeyse tüm derin modeller fay hattının konumunu doğru bir şekilde kestirebilmiştir. Ama fay hattının şekli ve atım boyu Model-I ve Model-II’nin hatta bazen Model-III’ün sonuçlarında doğru bir şekilde oluşturulamamıştır. Tabaka sayısının artması fay hattının kestirilmesini özellikle daha derin tabakalarda zor hale getirmektedir. Faylı modellerde ara yüzeylerin dalgalı veya kıvrımlı olması fay hattının kestirilmesini daha da zorlaştırmaktadır. Şekil 5.7’nin üçüncü satırındaki hız modeli bunun bir örneğidir.

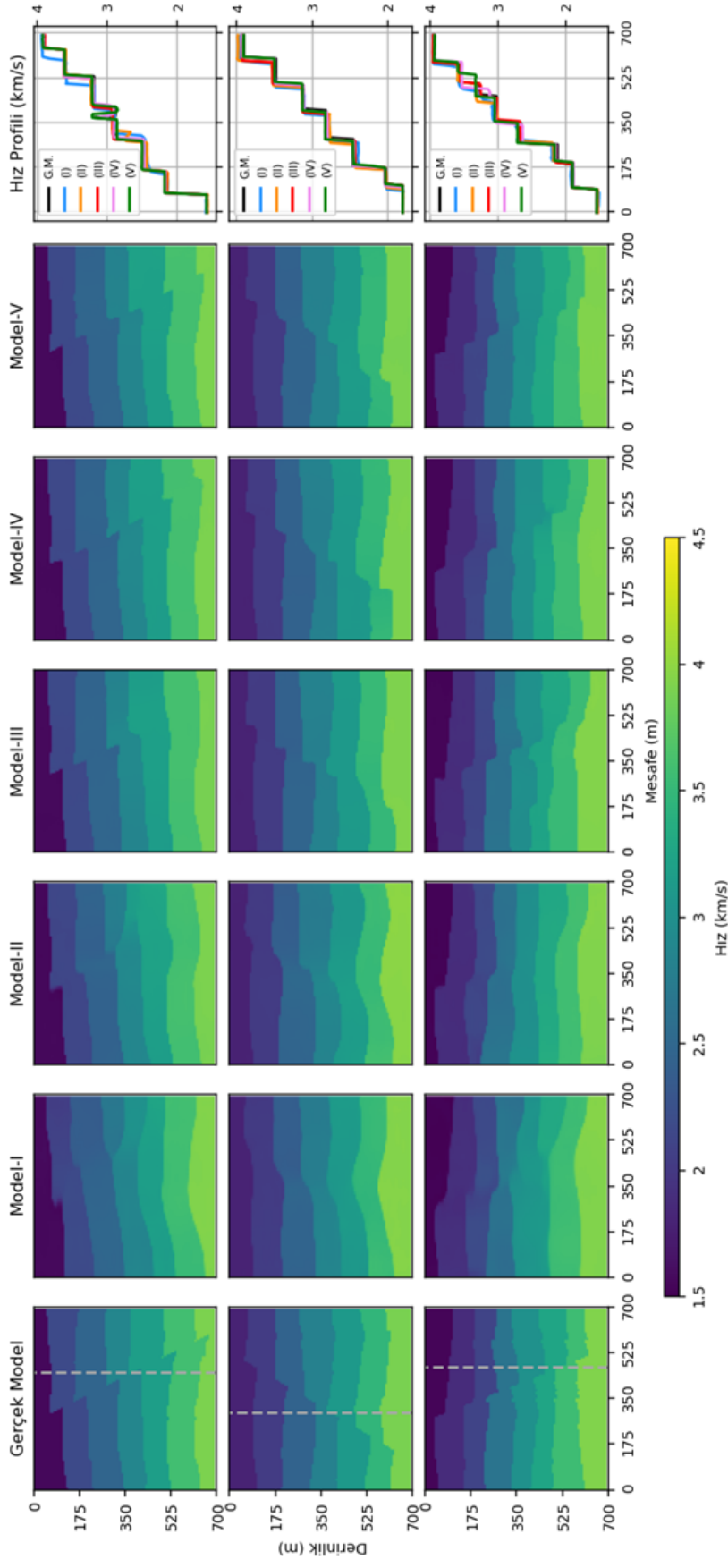
Tuz domu içeren hız modellerde (Şekil 5.8) ara yüzey ve tabaka şeklinin oluşturulması ile birlikte tuz domunun şekli ve konumu da eklenmektedir. Eğitim veri kümesi arttıkça tuz domunun sokulduğu tabakaların ara yüzeylerine göre konumu ve gövdesinin boyutu gerçek hız modeline yakın bir şekilde kestirilmektedir. Eğitilmiş derin modeller tuz domunun etkisi nedeni ile antiklinal şeklinde yukarıya doğru kıvrılan ara yüzeyleri tuz domunun kendisinden ayırt ederek üst tabakaların şeklini de kestirebilmeleri gerekmektedir.

Şekillerdeki hız profilleri gerçek hız modeli ile derin modelin kestirimlerini dişey bir hat boyunca detaylı olarak karşılaştırma amacıyla çizilmiştir. Eğitim verisi büyüdükçe tabakaların hız değeri ve tabakalar arası geçişler gerçek modele yakınsamaktadır. Ancak daha detaylı bir karşılaştırma amacı ile test veri kümesinden rastgele seçilen tabakalı, faylı ve tuz domu içeren örnekler için gerçek model ile derin modelin kestiriminin farkı da hesaplanmıştır. Şekil 5.9’da Model-I, Model-III, ve Model-V ait kestirimler ve fark kesitleri çizilmiştir. Tüm fark kesitleri eşit değerler aralığına ölçeklenmiş ve renk skalası hız modellerin renk skalasının sağ tarafında çizilmiştir.

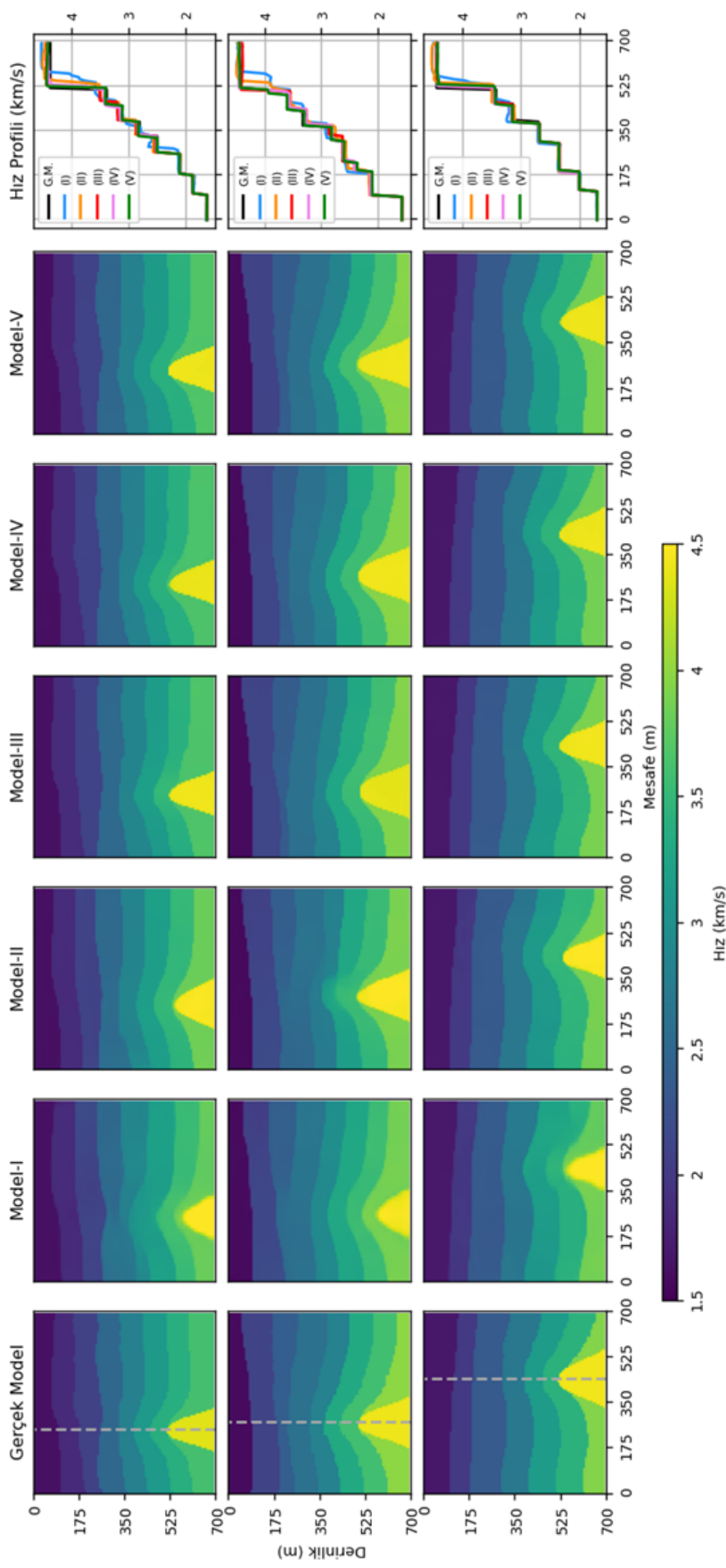
Fark kesitlerinde görüldüğü gibi Model-I ait kestirimlerde tabaka arası hız değerleri, arayüzeylerin şekli, tuz domunun yanal konumu ve kütesinin şekli gerçek hız modeli ile farklılık göstermektedir. Ancak gerçek hız modeli ile Model-V’in kestirimleri geçiş sınırlarında (arayüzeyler, fay hattı ve tuz domu sınırı) kesik çizgiler şeklinde görünmektedir.



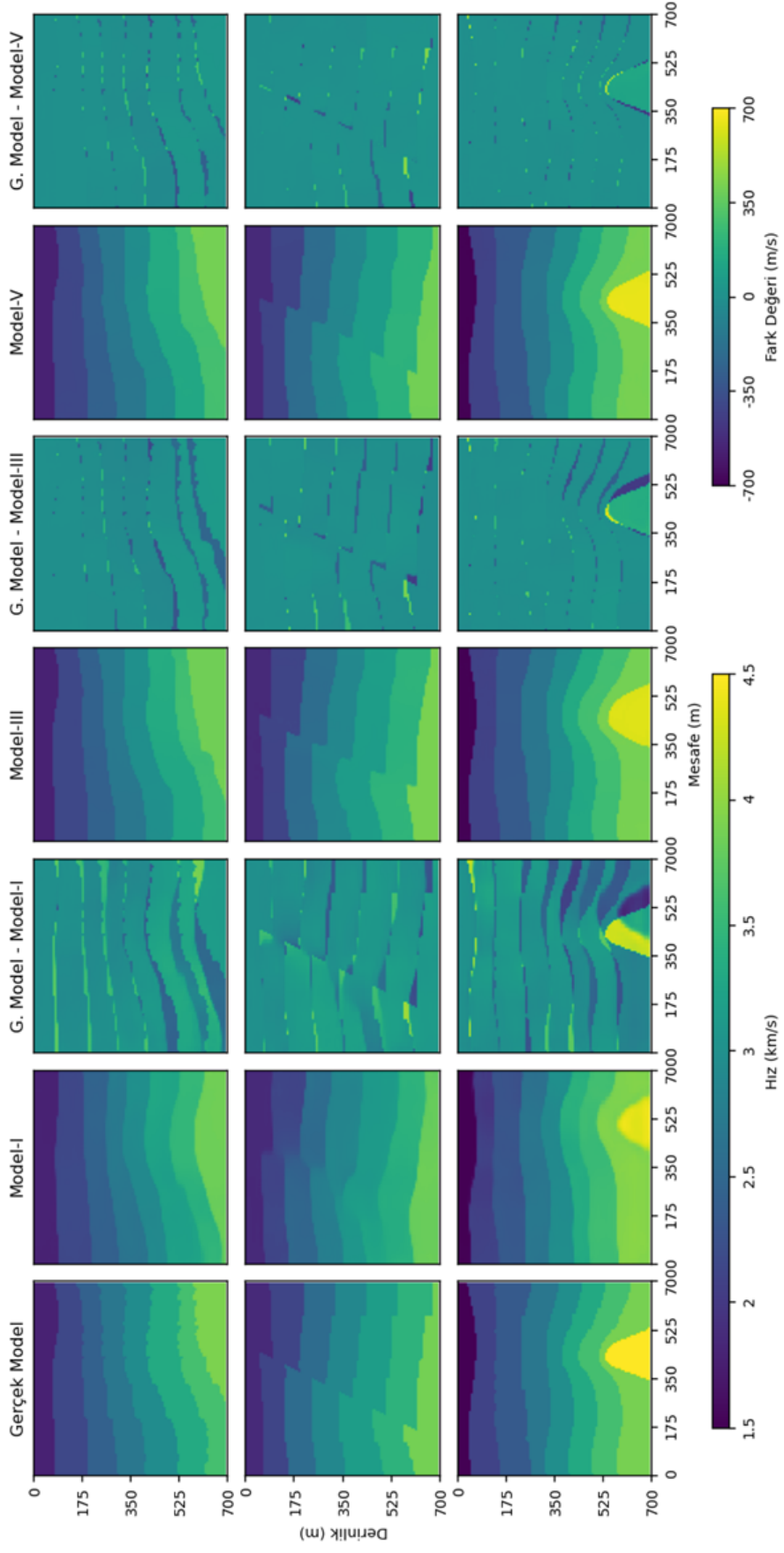
Şekil 5.6 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirdiği hız modelleri. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tabakalı modellerden rastgele seçilmiştir



Şekil 5.7 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in sonuçları. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece farklı modellerden rastgele seçilmiştir



Şekil 5.8 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirimi. Eğitim için gürültüsüz veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tuz domu içeren modellerden rastgele seçilmiştir



Şekil 5.9 Gerçek hız modeli, Model-I, Model-III, ve Model-V'in kestirimleri ve fark kesitleri. Eğitim için gürültüsüz veri kümesi kullanılmıştır

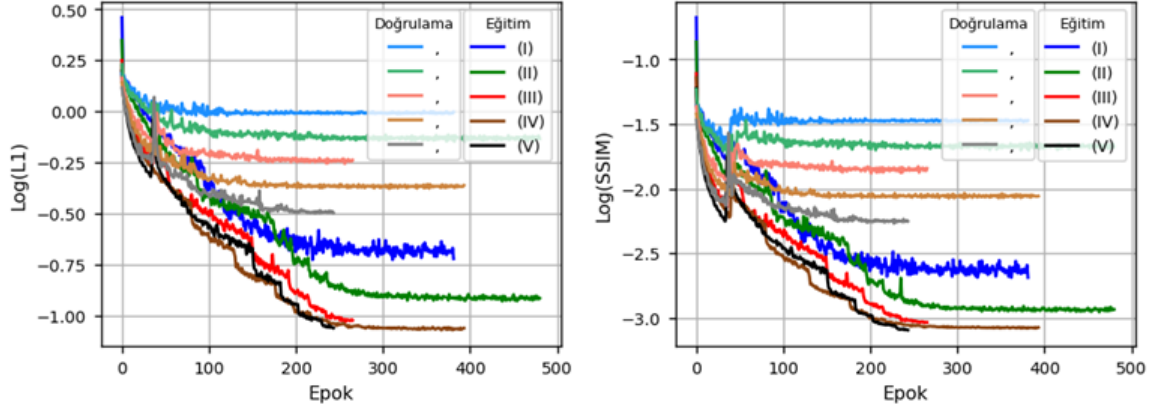
Gürültülü veri kümesine ait beş alt eğitim veri seti kullanılıp SVInvNet derin model ayrıca eğitilerek beş farklı eğitilmiş model (I-V) hazırlanmıştır. Gürültülü verilere ait test veri seti kullanılarak da her bir eğitilmiş modelin test aşaması gerçekleştirilmiştir. Test aşamasında elde edilen sayısal sonuçlar Çizelge 5.3'te sunulmaktadır. TD-V gürültülü eğitim veri kümesi ile eğitilen temel modelin, InversionNet( $\gamma$ )-V, test aşamasındaki sonuçları da karşılaştırma amacı ile çizelgenin son satırına eklenmiştir.

Gürültüsüz verilerde olduğu akım gürültülü verilerde de devam etmekte ve eğitim veri kümesinin artması hata değerlerinin, L1 ve L2, azalması ve benzerlik ölçütlerinin, SSIM ve MSSIM, büyümesi ile sonuçlanmaktadır. En iyi değerler Model -V'e aittir ve Model-IV performans olarak temel modeli geçmektedir. Ancak beklendiği gibi gürültülerin eklenmesi hata değerlerinin artmasına sebep olmaktadır. Çizelge 5.3 ile Çizelge 5.2'deki değerler kıyaslanınca gürültülü veri kümesi ile eğitilen modellerin hata değerlerinin neredeyse gürültüsüz verilerin iki kat büyük olduğu anlaşılmaktadır.

Gürültülü beş alt veri kümesi ile ayrı ayrı eğitilen SVInvNet derin modelin, Model I-V, eğitim ve doğrulama L1 ve SSIM yanılğı enerjilerine ait eğriler Şekil 5.10'da görülmektedir. Epok sayısının artması ile hata değerlerinin düşmesi bariz bir şekilde ayırt edilmektedir. Eğrilerin akımı, özellikle doğrulama verilerine ait yanılğı eğrileri (soluk renkler) dikkate alınırca, öğrenme işinin veri sayısından gürültüsüz senaryoya

Çizelge 5.3 SVInvNet derin modelin test aşamasındaki performansı. TD I-V gürültülü veri kümeleri kullanılarak sırası ile Model I-V eğitilmiştir. Son satırda InversionNet( $\gamma$ )-V'in test aşamasındaki sonuçları sunulmaktadır

	L1	L2	SSIM	MSSIM
Model-I	0.024961	0.001548	0.999856	0.999988
Model-II	0.018942	0.001044	0.999911	0.999992
Model-III	0.014189	0.000736	0.999945	0.999995
Model-IV	0.010378	0.000489	0.999967	0.999997
Model-V	<b>0.008034</b>	<b>0.000399</b>	<b>0.999977</b>	<b>0.999997</b>
InversionNet( $\gamma$ )-V	0.013142	0.000713	0.999949	0.999995



Şekil 5.10 SVInvNet derin modelin L1 ve SSIM yanlgı eneşisi eğrileri. Modeller I-V, sırasıyla gürültülü TD I-V kullanılarak eğitilmiştir

göre daha çok etkilendiğı anlaşılmaktadır. Eğitim yanlgı eğrileri hem L1 hem de SSIM için veri sayısının artması ile birlikte birbirine yanaşmalarına rağmen doğrulama yanlgı eğrileri belli bir mesafe ile bir alt değerde paralel olarak seyir etmektedir.

Gürültülerin etkisinin görsel olarak gösterilmesi için test veri denektaşından rastgele örnekler seçilerek beş eğitilmiş derin modelin ürettiğı sonuçlar Şekil 5.11- 5.14 sunulmaktadır. Sırası ile tabakalı, faylı ve tuz domu içeren hız modeli gruplarına ait olan bu şekilde de gerçek hız model ilk sütunda ve gerçek hız modeli üzerinde gri renkli kesik çizgi ile gösterilen şeritin hız profili son sütunda çizilmektedir. Eklenen gürültüler ile hız modellerin arasında herhangi bir ilişki olmadığı için sismik verilerle hız modelin arasındaki ilişkiyi öğrenmeye çalışan katsayıların doğru hesaplanmasını zorlaştırmaktadır.

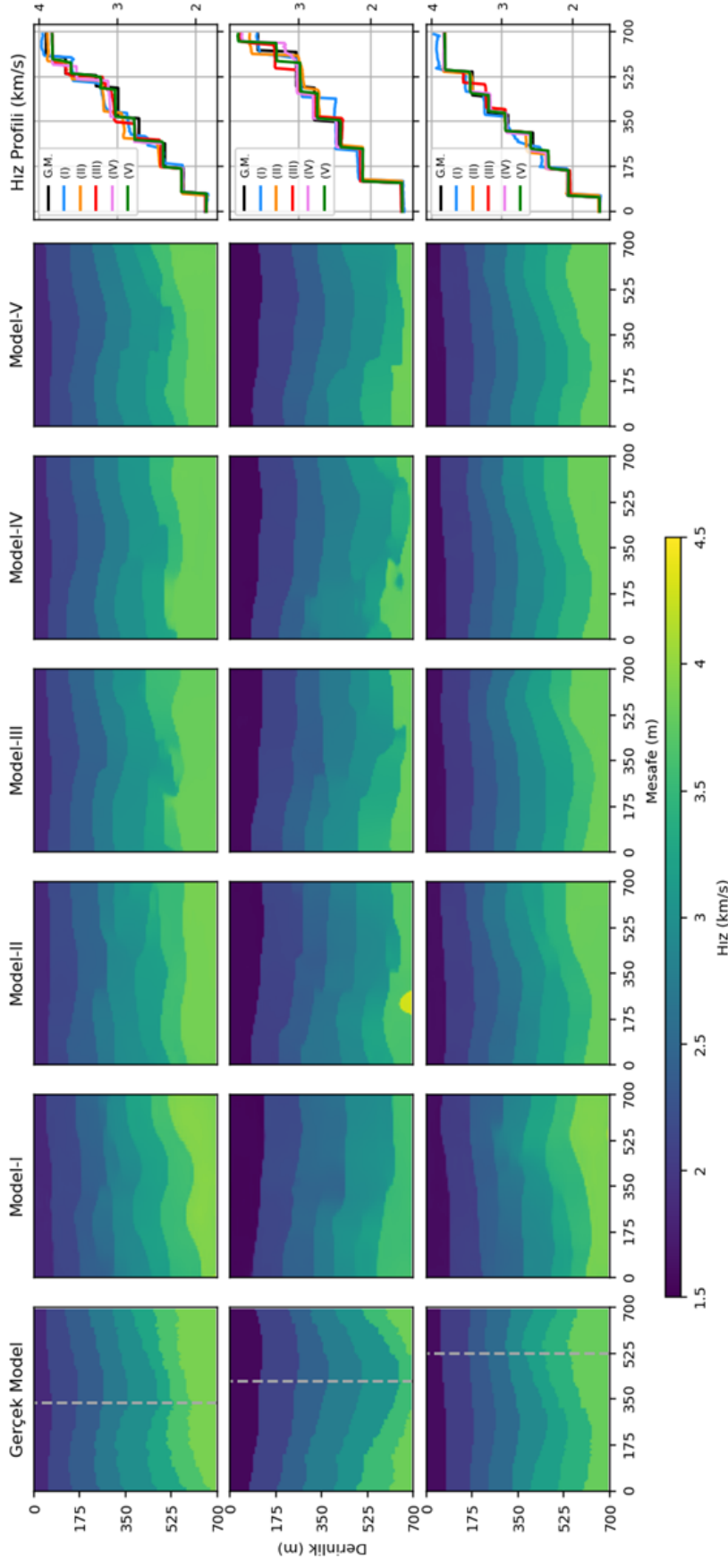
Tabakalı örnekleri gösteren Şekil 5.11'de olduğu gibi dalgalı veya yüksek eğimli ara yüzeyler derin model tarafından fay hattı veya bazen tuz domu olarak algılanmaktadır. Bu durum özellikle düşük sayılı eğitim veri kümesi ile eğitilen Model-I ve Model-II'de görünmektedir. İlk iki satırdaki örneklerde Model-V bile zorlanmakta ve daha iyi sonuçların elde edilmesi için eğitim veri kümesinin artırılması gerekmektedir.

Gürültüler sismik verilerdeki yansıma ve saçılma verilerini bastırarak faylı model-

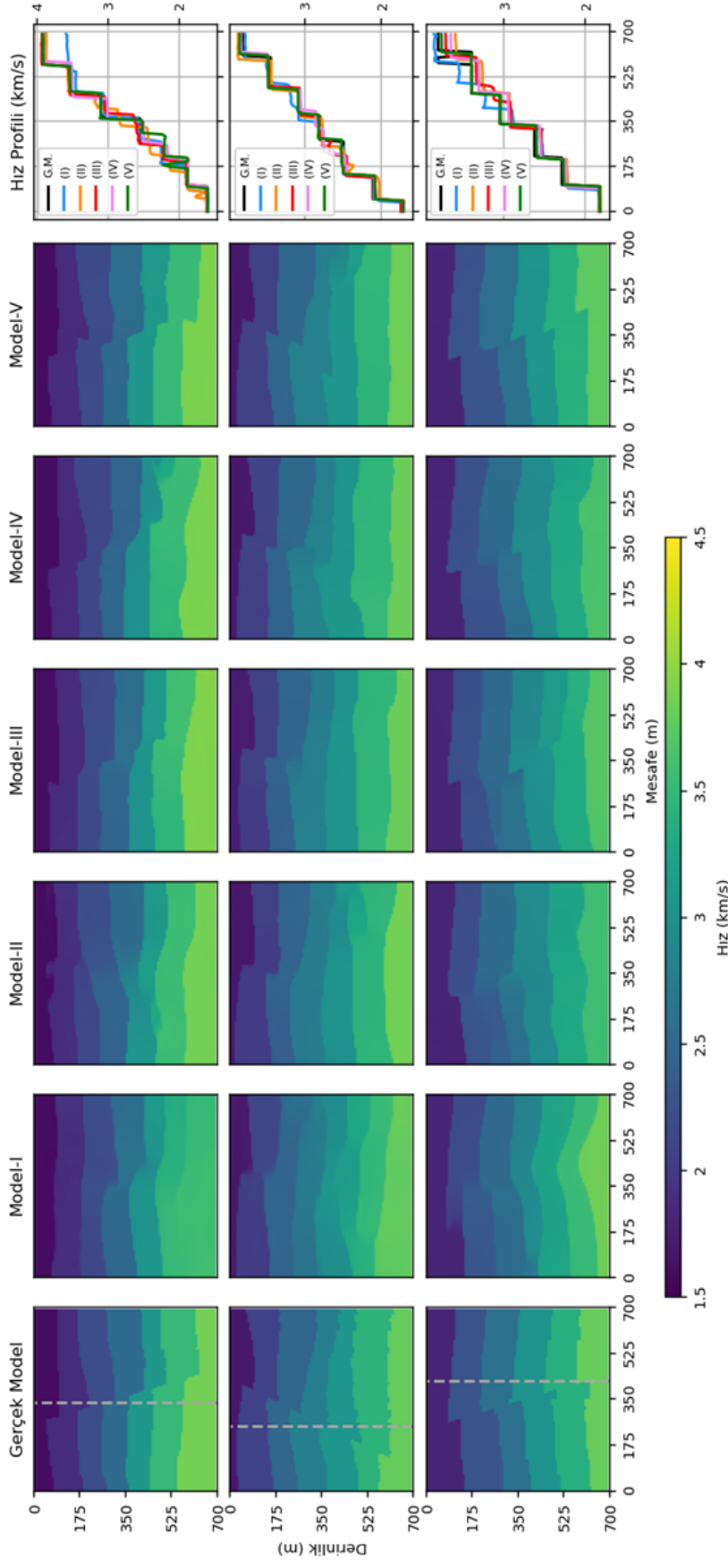
lerde fay hattının yönü, sürekliliği ve atım boyunun kestirilmesine mani olmaktadır. Şekil 5.12'deki örneklerde bu konu açık bir şekilde görünmektedir. Aynı zorluluk tuz domu içeren modellerde de geçerlidir. Üç farklı tuz domu modeli ve onlara ait derin modellerin kestirimi Şekil 5.13'te sunulmaktadır. Model-I ve Model-II'ye ait çıktılarda tuz domu şeklinin kestirilebilmesi için eğitim veri sayısının yetersiz olduğu anlaşılmaktadır.

Şekillerdeki örneklerin rastgele olarak seçilmesi vurgulanarak test veri kümesine ait hız modellerin içerisinde buradaki sonuçlardan daha iyi veya daha kötü kestirimlerin bulunabilmesi ifade edilmektedir. Gürültüsüz verilerde elde edilen sayısal ve nitel sonuçlara gürültülü verilerde de ulaşabilmek için eğitim veri kümesinin artması gerekmektedir. Çizelge 5.3 ve Çizelge 5.2'deki değerler dikkate alınarak bu artışın en az iki kat olması çıkarılmaktadır.

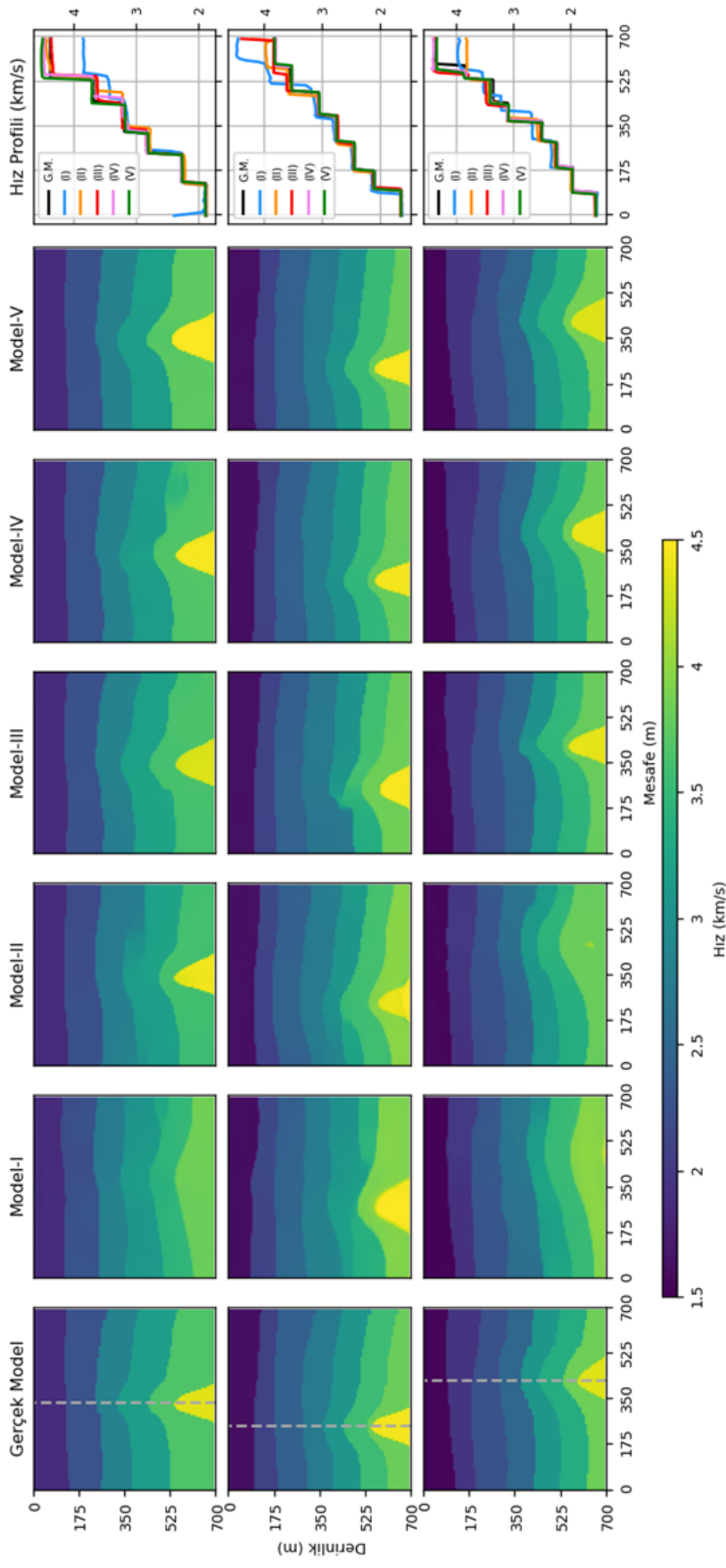
Gürültüsüz veriler senaryosunda olduğu gibi gürültülü veriler için de fark kesitleri Model-I, Model-III ve Model-V'in kestirimleri için hesaplanmıştır. Şekil 5.14'te gösterildiği gibi gürültülü verilerde daha büyük fark değerleri ortaya çıkmaktadır. Ama eğitim veri kümesinin artması burada da hata değerlerini geçiş sınırlarındaki detaylara indirgemektedir. Diğer bir deyişle gerçek hız modeli ile Model-V'in kestirimleri arayüzeyler, fay hattı ve tuz domunun sınırı boyunca farklılık göstermektedir.



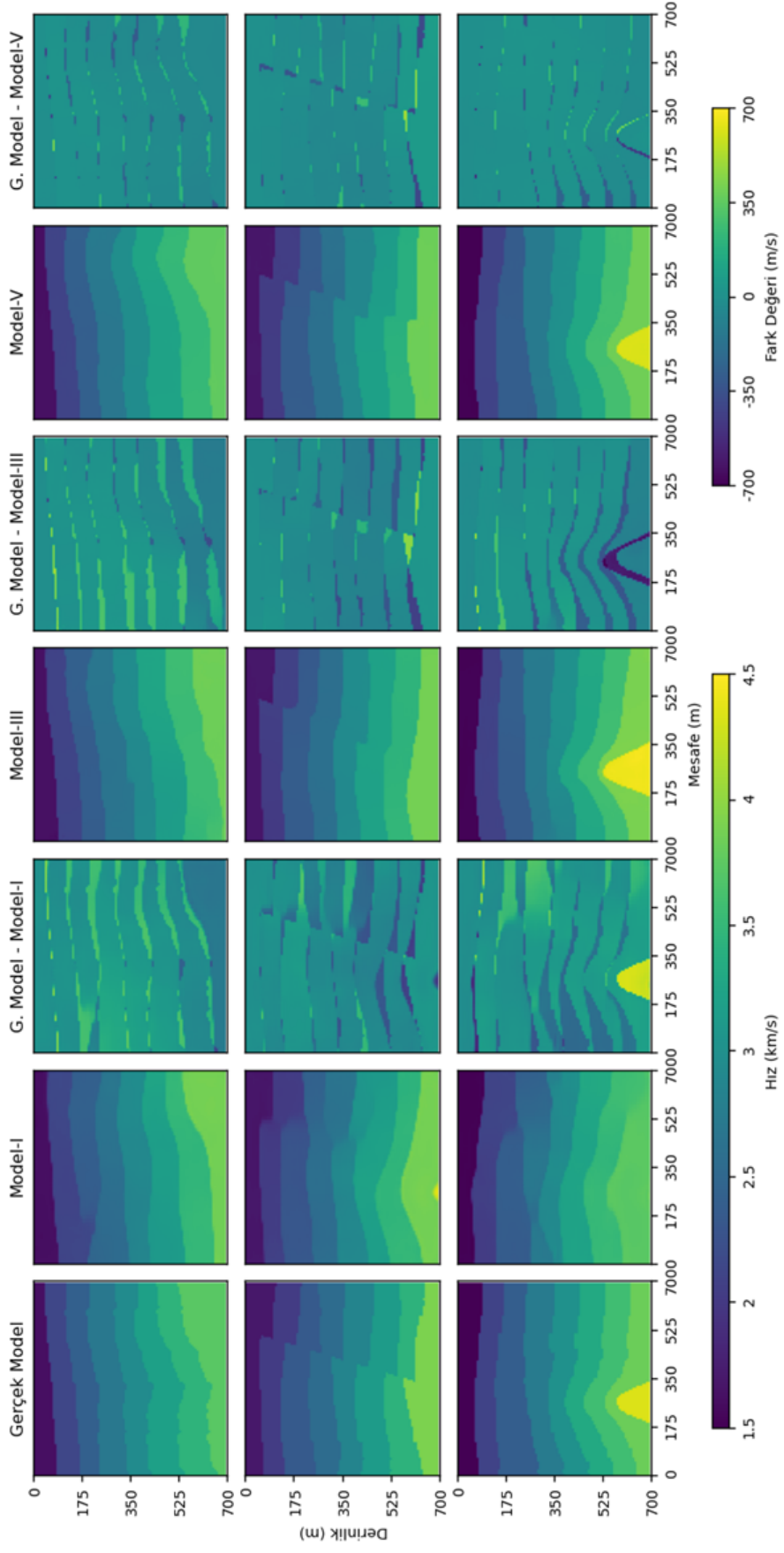
Şekil 5.11 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in kestirdiği hız modelleri. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tabakalı modellerden rastgele seçilmiştir



Şekil 5.12 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in ürettiği çıktılar. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece farklı modellerden rastgele seçilmiştir



Şekil 5.13 Gerçek hız modeli ve Model-I, Model-II, Model-III, Model-IV, ve Model-V'in sonuçları. Eğitim için gürültülü veri kümesi kullanılmıştır. Sağ sütunda gerçek modelde gri renkli kesik çizgi ile gösterilen şeritin hız profili yer almaktadır. Bu örnekler sadece tuz domu içeren modellerden rastgele seçilmiştir



Şekil 5.14 Gerçek hız modeli, Model-I, Model-III, ve Model-V'in kestirimleri ve fark kesitleri. Eğitim için gürültülü veri kümesi kullanılmıştır

### 5.3 SVInvNet - OpenFWI Veri Kümesi

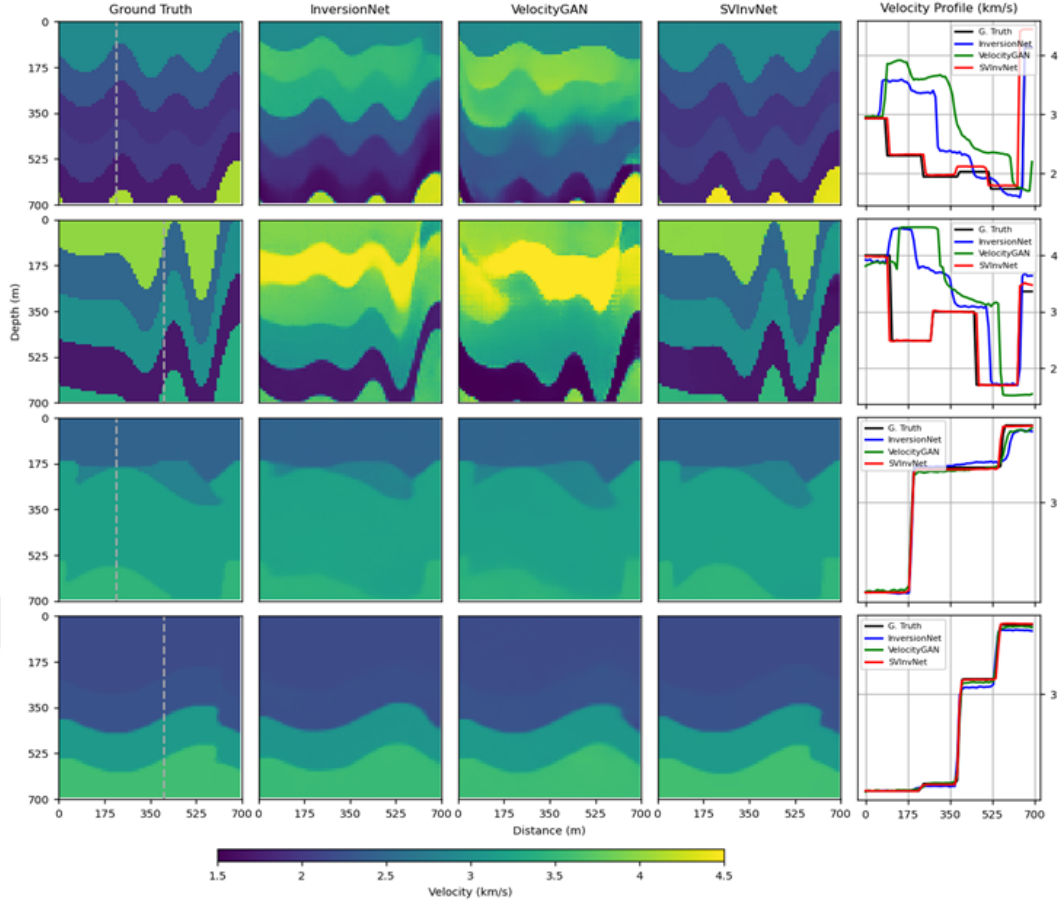
Bu tez kapsamında hazırlanmış olan eğitim veri kümesinden farklı olan OpenFWI veri kümesi kullanılarak SVInvNet derin modelin performansı denenmiş ve başka derin modellerle de karşılaştırılmıştır. Tabakalı ve faylı modeller olmak üzere sırasıyla CurveVel-B ve CurveFault-A OpenFWI veri kümesinin iki alt veri seti kullanılmıştır. Karşılaştırma için de aynı veri kümesi üzerinde eğitilmiş InversionNet, VelocityGan (Zhang ve Lin, 2020) ve UPFWI (Jin vd., 2022) derin modellerin Deng vd. (2022) tarafından raporlanan sonuçları kullanılmıştır.

Bu tez çalışmasında hazırlanan veri kümesinden farklı olarak OpenFWI veri setinde 70 alıcı vardır ve hız modellerin, çıktı verilerin, boyutu  $70 \times 70$  gritten oluşmaktadır. Derin modeli yeni veri seti ile uyumlu hale getirmek için SVInvNet'in ilk dört katmanına ek bir katman eklenip son katmanların boyutu da  $100 \times 100$  boyutundan  $70 \times 70$ 'e değiştirilmiştir. Eğitim ve doğrulama veri kümeleri Deng vd. (2022) yayınladığı şekilde ayarlanarak eğitim süreci gerçekleştirilmiştir.

SVInvNet derin model için test aşamasında elde edilen sonuçlar Deng vd. (2022) tarafından diğer üç derin model için yayımlanan sonuçlarla birlikte Çizelge 5.4'te sunulmaktadır. Bu çizelgede L2 (ortalama hata kareleri) yerine L2'nin kare kökü (Root Mean Squared Error - RMSE) hata ölçütü olarak kullanılmıştır. Çizelgedeki kestirim hatası ve benzerlik değerlerine dayanarak SVInvNet derin modelin kestirdiği hız modelleri diğer üç CNN modelinden daha iyi olduğu anlaşılmaktadır.

Çizelge 5.4 Test kümesindeki dört CNN Modelinin performansı. Modeller, CurveVel-B (CV-B) ve CurveFault-A (CF-A) veri setleri kullanılarak eğitilmiştir.

Ölçüt	CV-B			CF-A		
	L1	RMSE	SSIM	L1	RMSE	SSIM
SVInvNet	<b>0.0250</b>	<b>0.0583</b>	<b>0.9997</b>	<b>0.0042</b>	<b>0.0128</b>	<b>0.9999</b>
InversionNet	0.1448	0.3111	0.6630	0.0303	0.0766	0.9448
VelocityGAN	0.1268	0.2618	0.7111	0.0216	0.0505	0.9687
UPFWI	0.1777	0.3179	0.6614	0.0500	0.0966	0.9495



Şekil 5.15 Gerçek hız modelleri ve onlara ait InversionNet, VelocityGAN ve SVInvNet derin modellerin kestirimleri. İlk iki satır CurveVel-B ve son iki satır ise CurveFault-A benchmark'larından rastgele seçilmiştir. Hız profilleri, gerçek hız modellerinde çizilen ilgili gri çizgilere aittir

Deng vd. (2022) tarafından InversionNet ve VelocityGAN için yayınlanan eğitilmiş modellerden yararlanarak modellerin test aşamasında ürettikleri kestirimler arasında görsel karşılaştırma yapılmıştır (Şekil 5.15).

#### 5.4 FWI Yöntemi ile Karşılaştırma

Bu alt bölümde, faylı ve tuz domu içeren iki hız modeli test veri kümesinden rastgele seçilerek Model-I ve Model-V'in sonuçları ile türev tabanlı FWI yönteminin kestirimi karşılaştırılmıştır. Model-I ve Model-V sırayla en küçük ve en büyük eğitim veri kümesi ile eğitilmiş derin modeller olarak önceki alt bölümlerde yer alan sonuçlara göre sırayla en kötü ve en iyi sonuçları üreten modellerdir. Böylece FWI yöntemi

SVInvNet derin modelin en iyi ve en kötü performansı ile karşılaştırılmış olmaktadır.

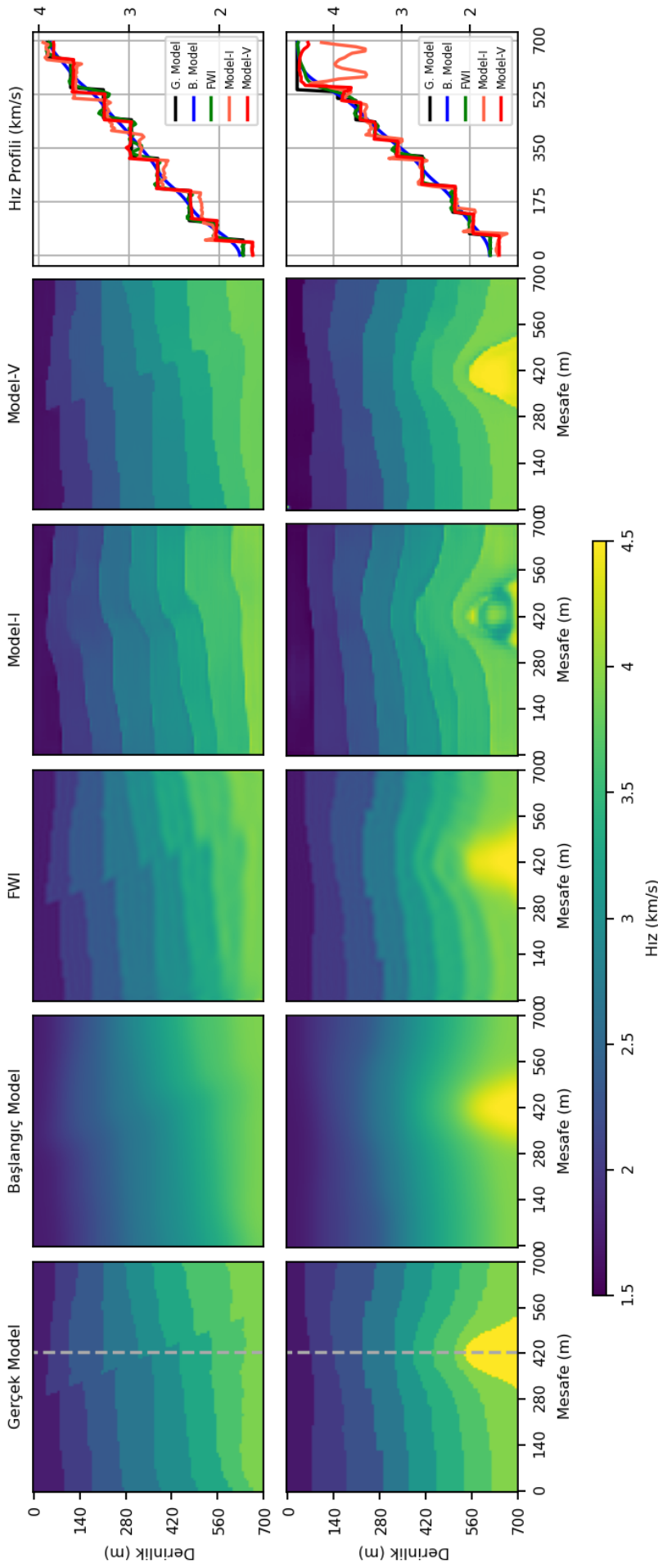
FWI yönteminin gürültüye yüksek duyarlılığı nedeniyle, hız modelleri gürültüsüz veri setlerinden seçilmiştir. Başlangıç hız modeli, gerçek modele standart sapması 5 olan Gauss yumuşatma fonksiyonu uygulanarak elde edilmiştir. Döngü atlama (cycle skipping) etkisini en aza indirmek amacıyla, işlem 10, 15, 20, 25, 30 Hz beş kesme frekansı seçilerek yürütülmüştür. Düşük frekanslardan başlayarak yüksek frekanslara doğru ilerleme tekniği Bunk vd. (1995) tarafından sunulmuştur. Ayrıca, FWI yönteminin etkinliğini artırmak için hız parametreleri 1000 – 5000 m/s arasında sınırlandırılarak çözüme bir koşul eklenmiştir. Gerçek hız modelinde hız değerleri 1500 – 4550 m/s aralığında değişmekte olduğu için bu koşul çözümü olumlu yönde etkilemektedir.

FWI'nin düz çözüm aşamasında kullanılan kurulum eğitim veri kümesinde sismik atış kesitlerinin hesaplaması için kullanılan parametreler, alıcı ve kaynak düzeni ile tutarlı kalmıştır. Yöntemin uygulanması için pytorch kütüphanesi kullanılarak tasarlanan DeepWave (Richardson, 2023) paketi kullanılmıştır. Bu pakette geri yayılım algoritması kullanılarak hesaplama maliyeti düşürülmüştür. FWI yöntemi yenilemeli adımlarla başlangıç hız modelin parametrelerini değiştirerek elde olan sismik atış kesitleri tekrar üretmek için değiştirmiştir. Bu işlem toplamda 50 iterasyon boyunca devam etmiştir.

Şekil 5.16'da, FWI, Model-I ve Model-V'in sonuçları gösterilmektedir. Üç sonucun gerçek hız modeli ile nitel kıyaslaması en iyi kestirimin Model-V'e ait olduğunu ispatlamaktadır. FWI yöntemi iyi bir başlangıç modelinden yola çıkmasına rağmen, daha derin tabakalardaki ara yüzlerin şeklini tahmin etmede zorluklarla karşılaşmaktadır. Ayrıca, FWI'nin kestiriminde tabakalar arası hız değeri sabit bir değere yaklaşmaktansa salınımlar ve sapmalar göstermektedir. Hız profili kesitinde, FWI sonuçlarını temsil eden yeşil çizgi, Model-V sonuçlarını gösteren kırmızı çizgiye kıyasla daha fazla kararsızlık ortaya koymaktadır.

Faylı hız modelin kırılma konumu ve hattı üretilen her üç sonuçta oluşturulmuş olsa da FWI modelin sol kenarını ve son ara yüzeyini bulanık bir şekilde kestirebilmiştir. Bu uyuşmazlıklar Model-V'in kestiriminde gözükmemekte ancak ufak farklar ayırt edilebilmektedir. Örnek için son ara yüzeyide fayın atım boyu doğru bir şekilde oluşturulamamıştır. Tuz domlu hız modeli için kullanılan başlangıç modelde tuz domun konumu ve dağılımı ile ilgili etraftaki tabakalardan ayırt edilebilir derecede bilgi yer alsa da, Model-V, FWI yöntemine göre tuz kütlelerinin hem şeklinin hem de hız değerlerinin oluşturmasında üstün performans sergilemektedir.

Şekil 5.16'daki sonuçların içinde en kötü tahmin Model-I'e aittir. Fay hattı gerçek modelde olduğu gibi keskin değildir ve tuz domunun şekli ve hız değeri kestirilememiştir. Ayrıca Tuz domlu modelde altıncı ara yüzey ve altıncı ve yedinci tabakaların hız değerleri neredeyse oluşturulamamıştır. Ancak bunlara rağmen sadece 750 adet örnek sayısı ile eğitilmiş derin modelin sunduğu sonuç gerçek hız modeli ile karşılaştırılınca iyi bir başlangıç modeli olarak FWI yöntemi için kullanılabilir olduğu ortadadır. FWI yönteminde doğru bir başlangıç modelin çözüm sürecindeki etkisi dikkate alınarak eğitilmiş derin modelin en kötü sonucunun bir başlangıç modeli olarak kullanılabilirliği başarı sayılmaktadır.

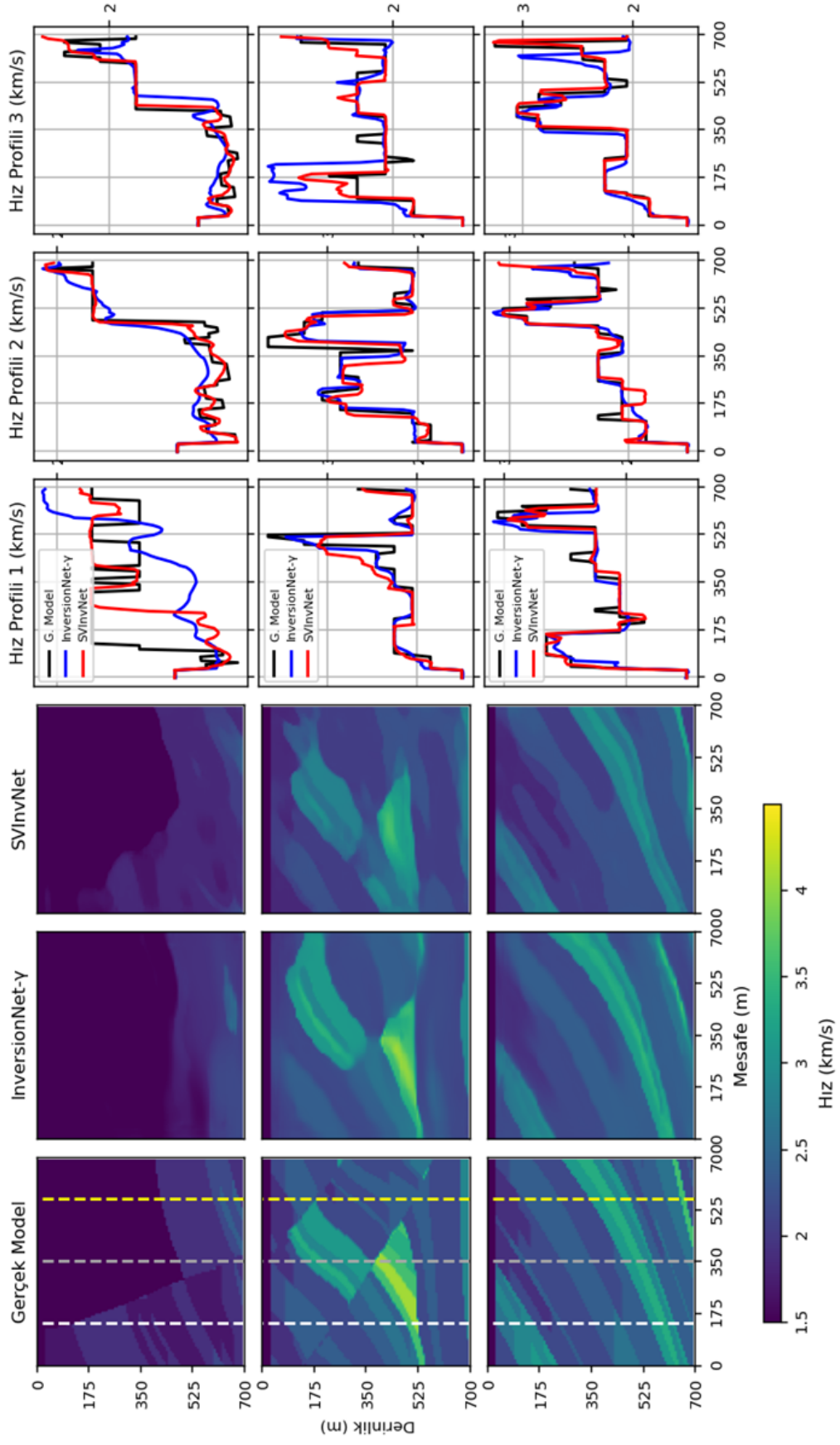


Şekil 5.16 FWI, Model-I ve Model-V'ın kestirdiği hız modelleri gösterilmektedir. Gerçek model dikkate alınarak en iyi sonuç Model-V'e aittir. FWI yönteminde kullanılan başlangıç hız modeli de sunulmaktadır

## 5.5 Marmousi Tabanlı Hız Modelleri

Bu alt bölümde SVInvNet ve InversionNet- $\gamma$  derin modellerin Marmousi hız modelinden alınan örnekler üzerindeki performansı karşılaştırılmıştır. Eğitim aşamasında Marmousi hız modelinden alınan alt hız modellerinin TD-V veri kümesine eklenmesi ile hazırlanan eğitim veri kümesi kullanılmıştır. Test aşamasında ise Marmousi hız modelinden alınan 23 test veri kümesi kullanılarak eğitilmiş derin modellerin daha gerçekçi hız modelleri için sundukları sonuçlar karşılaştırılmıştır.

Test aşamasında derin modellerin kestirdiği hız modellerinden rastgele olarak seçilen örnekler Şekil 5.17’de sunulmuştur. Şekilde görüldüğü üzere, SVInvNet’in kestirimleri temel modele kıyasla gerçek hız modeline daha yakındır. Her bir örneğin üç noktasından alınan hız profilleri daha detaylı bir nitel karşılaştırmaya imkan sağlamaktadır. Ancak eğitim veri seti, yüksek eğimli fay geçişleri ve düşük kontrastlı iç içe geçmiş eğimli katmanların doğru bir şekilde tahmin edilmesi için yetersiz kalmaktadır.



Şekil 5.17 Gerçek hız modeli, SVInvNet ve InversionNet- $\gamma$  derin modellerin kestirimleri. Hız profilleri 1-2-3 sırası ile gerçek modelde beyaz, gri ve sarı renkli kesik şekilde çizilen şeritlere aittir

## 6. TARTIŞMA VE SONUÇLAR

Eđitilmiş derin modellerin test veri kümesi üzerindeki performans deęerlendirilmesi nicel olarak üretilen hata ve benzerlik deęerlerine ve nitel olarak da kestirilen hız modelleri ile gerçek modellerin görsel karşılaştırılmasına dayanılarak yapılmaktadır. Kestirilen hız modellerinin görsel karşılaştırılmasında tabakaların hız deęeri, ara yüzeylerin şekli, fay hattının konumu, süreklilięi ve atım boyu, tuz domlarının şekli, konumu ve hız deęeri dikkate alınmalıdır. Bu parametreler hazırlanmış olan hız modellerin her üç grubu (tabakalı, faylı ve tuz domlu modeller), CurveVel-B ve CurveFault-A modelleri için geçerli olmaktadır.

Derinlięin artması ile hız deęerlerin ve ara yüzeylerin kestirilmesindeki hata deęeri büyümektedir. Derinlikle birlikte tabaka sayısı da artınca bu konu daha da karmaşık hale gelmektedir. Jeofizik bakış açısından yeraltına gönderilen sismik dalga alanı farklı tabakaların geçiş yüzeyinde yansıma, kırılma ve saçılma nedeni ile enerjisinin ve dolayısı ile genlięinin bir kısmını kaybetmektedir. Ayrıca dalga alanının daha derine inmesi enerjinin daha büyük bir yüzeye yayılması anlamına gelmekte ve bu ise başı başına bir sönüm etkenidir. Enerji kaybı genlik azalmasına sebep olmakta ve böylece çok tabakalı hız modellerinde ara yüzeylerin derinlięi artınca sismik atış kesitindeki ilgili yansıma veya saçılma hiperbollerini daha düşük genliklerle ortaya çıkarmaktadır.

Sismik atış kesitlerindeki yansıma, saçılma ve kırılma hiperbollerini etkileyen dięer bir etken ise yansıma katsayısıdır. Dalga alanının enerjisinin büyük kısmı yansıma katsayısı küçük olan ara yüzeylerden geçerek alt tabakalara inmekte ve küçük bir kısmı geri yansıyarak sismik kesitte ortaya çıkmaktadır. Yansıma katsayısı yüzeye yakın arayüzeylerin yansıma hiperbollerini de belli bir ölçüde etkilemektedir. Böyle durumlarda da kestirilen hız modellerin hata payı özellikle tabakaların oluşturulmasında büyük olmaktadır.

Faylı modellerde ara yüzeylerin fay hattı boyunca hareket etmesi keskin köşelerin ortaya çıkmasına sebep olmaktadır. Atım boyu büyük olan faylarda keskin köşeler daha da büyük ve belirgin şekilde görünmektedir. Keskin köşeler sismik dalga alanının saçılmasına dolayısıyla da ara yüzeylerin yansıma hiperbolünün sürekliliğinin bozulmasına neden olmaktadır. Daha sık ara yüzeylerde ortaya çıkan saçılma eğrileri alt tabakalardaki saçılma ve yansıma eğrilerini etkileyerek genliklerini bastırmaktadır. Hız modelin kenar sınırlarına yakın olan faylarda sismik dalga alanı fayın kenara yakın olan kısmını yeterince tarayamadığından bilgi eksikliği nedeniyle hız modellerin doğru bir şekilde kestirilmesi daha da karmaşık hale gelmektedir.

Tuz domları hız değeri ve şekil açısından etraftaki sedimentlerden veya tabakalardan farklı olmaktadır. Tuz domun uç noktasına erişen dalga alanı saçılarak sismik kesitte etrafındaki verilerle kıyasta şekil olarak belirgin bir saçılma hiperbolüne neden olmaktadır. Tuz domları hız modellerinin en alt tabakasında yer aldıkları için üst tabakaların yansıma verileri ile kıyasta genlik değerleri çok daha küçük olmaktadır. Tuz domunun sağ ve sol kenarlarından olası yansımalar da şekil ve hız değerinin kestirilmesi için yararlı bilgiler içermektedir. Bu nedenle de domun yanal konumu önem kazanmaktadır. Diğer bir deyişle hız modelin ortasına yakın olan tuz domları ile ilgili sismik atış kesitlerinde daha çok bilgi yer almakta ve bu ise derin modelin işini kolaylaştırmış olmaktadır.

Önceki bölümlerde bahsedilen konulara göre CNN çekirdekleri zaman eksenini boyunca küçültülen imgeleri kullanarak hız modelin oluşturmasını sağlayabilen özellikleri öğrenmektedir. Sismik verilere gürültünün eklenilmesi özneliklerin öğrenilmesini daha da zorlaştırmaktadır. Eğitim sürecinde CNN çekirdekleri hız modellerini daha düşük hata payı ile oluşturabilmek için gürültüleri verilerden ayırt edebilmeleri de gerekmektedir. Bu ise hız modelinin oluşturulmasıyla birlikte gürültü ayrıştırma işlemini de içermektedir.

Gürültülü verilerde aslında derin model aynı zamanda iki iş yapmaktadır bir taraftan sismik veri ile hız modelin arasındaki ilişkiyi çıkarmaya diğer taraftan ise gürültüyü

sinyalden ayırt etmeye çalışmaktadır. Eğitim veri kümesinin sayısı eğitilmiş derin modelin gürültüyü sinyalden ayırt edebileceğini sağlayacak kadar büyük olması gerekmektedir. Hız modellerin kestiriminde gürültülü veriler ile de gürültüsüz verilerde elde edilen hata değerlerine ulaşabilmek için eğitim veri kümesi en az gürültüsüz veri kümesinin iki katı olması gerekmektedir.

Derin modele girdi olarak verilen bilgilerin hangi bölümü ve nasıl kullanıldığı ile ilgili kesin çıkarımlar bilimsel olarak daha detaylı denemeleri gerektirmektedir. Ancak sismik kesitlerde yansıma ara yüzleri belirgin hiperbollerle ortaya çıkan hız modellerin daha doğru kestirilmesi denetlenmiştir. Bu nedenle de derin modelin kullandığı bilgiler listesine varış zamanı ve sinyal genliğinin yer alması düşünülebilir. Ayrıca bahsedilen verilerin gürültüler tarafından bastırması sonucunda ortaya çıkan olumsuz etkiler de gözlemlenmiştir.

Rastgele gürültüler herhangi bir düzene uymaksızın farklı genliklerle sismik atış kesitin farklı noktalarında sismik verilerin genliğini etkileyerek yansıma ve saçılma (ve olası kırılma) eğrilerin bozulmasına neden olabilmektedir. Bu değişim özellikle sinyal-gürültü etkileşiminin ayrıştırılması zor olan durumlarda, örneğin ince tabakalarda, kestirilen hız modelin ara yüzey şeklini ve/veya hız değerini etkileyebilmektedir.

Tutarlı gürültüler düzen açısından her bir hız modeline ait atış kesitlerinde belli bir hız değeri ile ortaya çıkmaktadır. Tutarlı gürültülerin hız değerleri hız modellerin değerlerinden çok daha küçük olduğundan derin model tarafından ayrıştırılarak kestirilen hız modellerinde bir tabaka gibi değerlendirilmemektedir. Genlik değerleri zamanla azalmakta olan bu gürültüler derin tabakalardan gelen sismik genliklerini ve dolayısıyla yansıma ve saçılma eğrilerini etkileyebilmektedir. Bu konuda rastgele gürültülerde olduğu gibi değerlendirilebilmektedir.

Elde edilen sonuçlara göre InversionNet derin modelin kod çözücü birimine uygulanan değişiklik başarılı sonuçlara neden olmaktadır. Diğer bir deyişle ingelerin büyütülmesi için dekonvolüsyon yerine upsample + konvolüsyon katmanının kullanılması

bu problemde başarılı bir değişiklik olarak değerlendirilmektedir. Tez çalışmaları sürecinde yapılmış olan denemelerin sayesinde elde edilmiş olan bu sonuç SVInvNet derin modelin tasarımında da kullanılmıştır.

InversionNet derin modelin üç versiyonunun,  $(\alpha)$ ,  $(\beta)$  ve  $(\gamma)$ , gürültüsüz ve gürültülü veri kümeler üzerinde test aşamasındaki performansı InversionNet derin modelin mimarisinde yapılmış olan değişikliklerin olumlu etkisini göstermektedir. Bu tez çalışmaları kapsamında yapılmış olan değişiklikler sonucunda InversionNet derin modelin parametre sayısı 44 miyondan 20 milyona düşürülmesine rağmen modelin performansı daha da iyileşmiştir.

Her bir tabakanın hız değeri sabittir ve herhangi bir değişim göstermeyerek sadece ara yüzey sınırlarında ani geçişlerle değişmektedir. Sunulan örnek hız profillerinde görüldüğü gibi eğitim veri kümesinin artırılması kestirilen hız modellerinin tabakalar içi hız değerinin gerçek değerine yaklaşmasına, dalgalanmaların ve sapmaların azalmasına neden olmaktadır. Hız değerlerin doğru kestirilmesi tabaka şeklinin, üst ve alt sınır şekli dikkate alınarak, amaç hız modeline yaklaşması anlamına gelmektedir. Eğitim veri kümesinin artması ile yanlış enerjiinin azalması test veri kümeleri üzerinde elde edilen sayısal değerlerde de görülmektedir.

Gürültülü ve gürültüsüz veri kümeleri üzerindeki görsel ve sayısal performans değerleri dikkate alınarak SVInvNet derin model InversionNet $\gamma$ 'ya göre daha iyi sonuçlar üretmektedir. OpenFWI veri kümesi üzerindeki sonuçlara göre SVInvNet performans olarak InversionNet, VelocityGan ve UPFWI derin modellerini geride bırakmaktadır. Eşit denek taşları üzerinde elde edilen bu sonuçlar SVInvNet derin modelin mimari olarak daha gürbüz ve kararlı olduğunu göstermektedir.

SVInvNet derin model FWI yöntemine göre daha başarılı sonuçlar çok kısa zamanda sunmaktadır. Elde edilen sonuçlara göre en az veri kümesi ile eğitilen SVInvNet derin modelin kestirdiği hız modeli FWI yöntemi için en kötü durumda başlangıç hız modeli olarak kullanılabilir özelliktedir.

SVInvNet'in performansını deęerlendirmek amacıyla, yeraltı gerek hız daęılımını daha doęru temsil eden Marmousi modelinden elde edilen alt modeller kullanılmıřtır. SVInvNet modeli temel modele kıyasla (InversionNet- $\gamma$ ) daha stn bir performans sergilemektedir. zellikle genel yapıların tahmininde belirgin iyileřmeler gözlemlenmiřtir. Bununla birlikte, kullanılan eęitim veri setinin, yksek eęimli fay geiřleri ve dřk kontrastlı i ie gemiř eęimli tabakalar gibi daha karmařık yapısal zellikleri doęru řekilde tahmin etmek iin yeterince kapsamlı olmadıęı da grlmektedir. Bu durum, eęitim veri setinin eřitlilięinin ve rneklerin sayısının arttırılması gereklilięine iřaret etmektedir.



## KAYNAKLAR

- Adler, A., Araya-Polo, M., and Poggio, T. 2021. Deep learning for seismic inverse problems: Toward the acceleration of geophysical analysis workflows. *IEEE Signal Processing Magazine*, 38(2):89–119.
- Alford, R. M., Kelly, K. R., and Boore, D. M. 1974. Accuracy of finite-difference modeling of the acoustic wave equation. *Geophysics*, 39(6):834–842.
- Araya-Polo, M., Jennings, J., Adler, A., and Dahlke, T. 2018. Deep-learning tomography. In *Leading Edge*, 37, 58–66.
- Ben-Hadj-Ali, H., Operto, S., and Virieux, J. 2008. Velocity model building by 3d frequency-domain, full-waveform inversion of wide-aperture seismic data. *GEOPHYSICS*, 73(5):VE101–VE117.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. 2014. Learning phrase representations using rnn encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
- Clayton, R. W. and Engquist, B. 1980. Absorbing boundary conditions for wave-equation migration. *Geophysics*, 45(5):895–904.
- Cybenko, G. 1989. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314.
- Deng, C., Feng, S., Wang, H., Zhang, X., Jin, P., Feng, Y., Zeng, Q., Chen, Y., and Lin, Y. 2022. Openfwi: Large-scale multi-structural benchmark datasets for full waveform inversion. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, 35, 6007–6020. Curran Associates, Inc.
- Devaney, A. J. 1984. Geophysical diffraction tomography. *IEEE Transactions on Geoscience and Remote Sensing*, GE-22(1):3–13.
- Fukushima, K. 1980. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202.
- Gauthier, O., Virieux, J., and Tarantola, A. 1986. Two-dimensional nonlinear inversion of seismic waveforms: numerical results. In *Geophysics*, 51, 1387–1403.
- Goodfellow, I., Bengio, Y., and Courville, A. 2016. *Deep Learning*. Adaptive Computation and Machine Learning series. MIT Press.
- Huang, G., Liu, Z., van der Maaten, L., and Weinberger, K. Q. 2017. Densely connected convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 4700–4708.
- Huang, Z., Shimeld, J., Williamson, M., and Katsube, J. 1996. Permeability prediction with artificial neural network modeling in the venture gas field, offshore eastern canada. *GEOPHYSICS*, 61(2):422–436.

- Iversen, E. and Gjøystdal, H. 1996. Event-oriented velocity estimation based on prestack data in time or depth domain. *Geophysical Prospecting*, 44(4):643–686.
- LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., and Jackel, L. D. 1989. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551.
- Lewis, W. and Vigh, D. 2017. Deep learning prior models from seismic images for full-waveform inversion. *All Days:SEG–2017–17627643*.
- Li, S., Liu, B., Ren, Y., Chen, Y., Yang, S., Wange, Y., and Jiang, P. 2020. Deep-learning inversion of seismic data. In *IEEE Transactions on Geoscience and Remote Sensing*, 58, 2135–2149.
- Lines, L. R., Slawinski, R., and Bording, R. P. 1999. A recipe for stability of finite-difference wave-equation computations. *Geophysics*, 64(3):967–969.
- Lippmann, R. P. 1988. An introduction to computing with neural nets. 16(1):7–25.
- Liu, B., Yang, S., Ren, Y., Xu, X., Jiang, P., and Chen, Y. 2021. Deep-learning seismic full-waveform inversion for realistic structural models. In *Geophysics*, 86, R31–R44.
- Maurya, S., Singh, N., and Singh, K. 2020. *Seismic Inversion Methods: A Practical Approach*. Springer Geophysics. Springer International Publishing.
- McCormack, M. D., Zaucha, D. E., and Dushek, D. W. 1993. First-break refraction event picking and seismic data trace editing using neural networks. *GEOPHYSICS*, 58(1):67–78.
- McCulloch, W. S. and Pitts, W. 1943. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133.
- MURAT, M. E. and RUDMAN, A. J. 1992. Automated first arrival picking: A neural network approach. *Geophysical Prospecting*, 40(6):587–604.
- Najafi Khatounabad, M., Yalim Keles, H., and Kadioglu, S. 2025. Svinvnet: A densely connected encoder-decoder architecture for seismic velocity inversion. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–12.
- Pan, G. S., Phinney, R. A., and Odom, R. I. 1988. Full-waveform inversion of plane-wave seismograms in stratified acoustic media: Theory and feasibility. *GEOPHYSICS*, 53(1):21–31.
- Poulton, M. M., Sternberg, B. K., and Glass, C. E. 1992. Location of subsurface targets in geophysical data using neural networks. *GEOPHYSICS*, 57(12):1534–1544.
- Roethe, G. and Tarantola, A. 1991. Use of neural networks for inversion of seismic data. *SEG Tech. Program Expanded Abstr.*, 302–305.

- Rosenblatt, F. 1958. The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.
- Rumelhart, D. E., Hinton, G. E., and Williams, R. J. 1986. Learning representations by back-propagating errors. *nature*, 323(6088):533–536.
- Schuster, G. 2017. *Seismic Inversion. Investigations in Geophysics.* Society of Exploration Geophysicists.
- Sen, M. K. and Stoffa, P. L. 1992. Rapid sampling of model space using genetic algorithms: examples from seismic waveform inversion. *Geophysical Journal International*, 108(1):281–292.
- Shrestha, A. and Mahmood, A. 2019. Review of deep learning algorithms and architectures. *IEEE Access*, 7:53040–53065.
- Simon, J., Fabien-Ouellet, G., and Gloaguen, E. 2024. Quality control in deep learning and confidence quantification: Seismic velocity regression through classification. *Computers Geosciences*, 184:105534.
- Sirgue, L. and Pratt, R. G. 2004. Efficient waveform inversion and imaging. a strategy for selecting temporal frequencies. In *Geophysics*, 69, 231–248.
- Stefani, J. P. 1995. Turning-ray tomography. *GEOPHYSICS*, 60(6):1917–1929.
- Tappert, C. C. 2019. Who is the father of deep learning? In *2019 International Conference on Computational Science and Computational Intelligence (CSCI)*, 343–348.
- Tarantola, A. 1984. Inversion of seismic reflection data in the acoustic approximation. In *Geophysics*, 48, 1259–1266.
- van der Baan, M. and Jutten, C. 2000. Neural networks in geophysical applications. *Geophysics*, 65(4):1032–1047.
- Versteeg, R. 1994. The marmousi experience: Velocity model determination on a synthetic complex data set. *The Leading Edge*, 13(9):927–936.
- Virieux, J. and Operto, S. 2009. An overview of full-waveform inversion in exploration geophysics. In *Geophysics*, 47, WCC127–WCC152.
- Wang, Y. 2016. *Seismic Inversion: Theory and Applications.* Wiley.
- Wang, Z. and Nur, A. 1990. Wave velocities in hydrocarbon-saturated rocks: Experimental results. *Geophysics*, 55(6):723–733.
- Warner, M., Ratcliffe, A., Nangoo, T., Morgan, J., Umpleby, A., Shah, N., Vinje, V., Štekl, I., Guasch, L., Win, C., Conroy, G., and Bertrand, A. 2013. Anisotropic 3d full-waveform inversion. *GEOPHYSICS*, 78(2):R59–R80.
- Williamson, P. R. 1991. A guide to the limits of resolution imposed by scattering in ray tomography. *GEOPHYSICS*, 56(2):202–207.

- Wu, Y., Lin, Y., and Zhou, Z. 2018. Inversionnet: Accurate and efficient seismic waveform inversion with convolutional neural networks. In In Proc. SEG Tech. Program Expanded Abstr, 2096–2100.
- Yang, F. and Ma, J. 2019. Deep-learning inversion: A next-generation seismic velocity model building method. *Geophysics*, 84(4):R583–R599.
- Zhang, Y. and Paulson, K. V. 1997. Magnetotelluric inversion using regularized hopfield neural networks. *Geophysical Prospecting*, 45(5):725–743.



## EK1 Akustik Ortamda Sismik Dalga Yayılımı

Sıvı veya gaz halinde olan maddeler dışarıdan uygulanan kuvvete karşı iç kuvvetlerle tepki göstererek boyutlarını korumaya çalışmaktadır. Dış kuvvet ortadan kalkınca da elastik özelliklerinin sayesinde eski hallerine geri dönmektedir. Bu maddelerde şekil özelliği söz konusu olmadığına göre iç kuvvetler dış kuvvetlerle aynı doğrultuda ama ters yönde oluşmaktadır. Örneğin  $1B$  ortamda ani bir dürtü nedeni ile oluşan dalga alanı dürtü yönünde ve partiküller ise dürtü doğrultusunda hareket etmektedir. Bu bir akustik ortamı tanımlıdır.

Elastik özellikleri açısından akustik ortamda sadece Bulk (yığımsa) modülü tanımlanmakta ve diğer modüller (Young ve Poisson) sıfıra eşdeğerdir. Makaslama gerilimi sıfır olduğu nedeni ile de akustik ortamda  $S$  dalga alanı oluşmamakta ve sadece  $P$  dalga alanı ve yayılım hızı tanımlanabilmektedir. Katı bir ortam için de akustik varsayımı ile  $P$  dalga alanının yayılımı ve seyahat zamanı yaklaşık olarak hesaplanabilmektedir. Akustik varsayımı ile hesaplanan dalga alanı yeraltındaki her bir noktanın yer değişimini skaler değer ile tanımlamaktadır. Dalga alanının zaman ile de değişmesi dikkate alınarak partiküllerin hareket fonksiyonu hem mekana hem de zamana bağlı bir değişken olarak iki boyutta  $P(x, z, t)$  şeklinde ifade edilmektedir.

Akustik basınç olarak da adlanan  $P$  değerini farklı zamanlarda,  $t$ , ve konumlarda,  $(x, z)$ , hesaplayan izotropik akustik dalga alanı yayılımı Denlem 1 ile tanımlanmaktadır. Bu ikinci dereceden kısmi diferansiyel denkleminde  $v$  ortamın faz hızı modelini ve  $S$  ise sismik kaynağını simgelemektedir. Bu denklem, ortamın yoğunluğu sabit ve homojen olduğu varsayımıyla türetilmiş olup, dalgaların zaman içindeki evrimini tanımlamaktadır.

$$\frac{\partial^2 P}{\partial t^2} - v^2 \left( \frac{\partial^2 P}{\partial x^2} + \frac{\partial^2 P}{\partial z^2} \right) = \mathcal{S}(t, x, z) \quad (1)$$

Denklem 1'in çözümünde farklı sayısal yöntemler kullanılmaktadır. Bu tez çalışmasında sonlu farklar açık (explicit) yöntemi kullanılarak hesaplama işlemi yapılmıştır. Bu yöntemde türevler, sonlu farklar kullanılarak ayrıklaştırılır. Zaman ve uzaysal türevleri için merkezi farklar yöntemi uygulanarak ikinci mertebeden doğrulukta bir ayrıklaştırma elde edilmiştir.

$$\frac{\partial^2 P}{\partial x^2} \approx \frac{P_{i+1,j}^l - 2P_{i,j}^l + P_{i-1,j}^l}{h^2} \quad (2)$$

$$\frac{\partial^2 P}{\partial z^2} \approx \frac{P_{i,j+1}^l - 2P_{i,j}^l + P_{i,j-1}^l}{h^2} \quad (3)$$

$$\frac{\partial^2 P}{\partial t^2} \approx \frac{P_{i,j}^{l+1} - 2P_{i,j}^l + P_{i,j}^{l-1}}{\Delta t^2} \quad (4)$$

Burada zaman eksenini  $\Delta t$  ve mekan eksenleri ise  $\Delta x$  ve  $\Delta z$  örnekleme aralıkları ile sayısallaştırılmış ve  $l$  zaman adımını,  $(i, j)$  ise uzaydaki düğüm noktalarını temsil etmektedir. Hesaplamada kolaylık olsun diye  $\Delta x = \Delta z = h$  şeklinde ayarlanmıştır. Ayrıklaştırılmış türevler, Denklem 2-4, akustik dalga denklemine, Denklem 1, yerleştirilerek sayısal çözüme uygun hale getirilmiştir (Denklem 5).

$$P_{i,j}^{l+1} = \frac{\Delta t}{v^2 h^2} (P_{i+1,j}^l + P_{i-1,j}^l + P_{i,j+1}^l + P_{i,j-1}^l - 4P_{i,j}^l) + 2P_{i,j}^l - P_{i,j}^{l-1} + S \quad (5)$$

İleri fark şeması ile zamansal evrimin sayısal olarak Denklem 5'te hesaplanmaktadır. Dalga yayılımının kararlılığını sağlamak için Courant-Friedrichs-Lewy (CFL) koşulu dikkate alınmalıdır. Denklem 6 ile ifade edilen bu koşula göre sayısal kararlılık doğru bir zaman adımı seçimi ile sağlanabilmektedir.

$$\frac{v\Delta t}{\min(\Delta x, \Delta z)} \leq 1 \quad (6)$$

Dalga yayılımının doğru modellenebilmesi için uzay adımları da bir kararlılık koşu-

lunu sağlamalıdır. uzaysal örnekleme koşulu Denklem 7 ile tanımlanmaktadır.

$$\lambda_{min} \geq 10 \min(\Delta x, \Delta z) \quad (7)$$

Burada  $\lambda_{min}$  en küçük dalga boyunu ifade etmekte ve uzaysal çözünürlüğün yeterli olmasını sağlamak için genellikle en az 10 örnekleme noktası gerekmektedir.

Çözümde yapay kaynak fonksiyonu olarak  $(x_0, z_0)$  noktasında Ricker dalgacığı kullanılmaktadır. Ricker dalgacığı zamandaki Gauss türevine dayalı olup izleyen şekilde ifade edilmektedir.

$$S(t) = (1 - 2\pi^2 f_0^2 t^2) e^{-\pi^2 f_0^2 t^2} \quad (8)$$

Burada  $f_0$  merkezi frekanstır. Bu dalgacık, özellikle sismik ve akustik dalga modellemelerinde sıkça tercih edilmektedir.