Yapay Zekaya Giriş — Derin Öğrenme Projesi Raporu

Beyin MRI Görüntülerinden Tümör Tespiti (Binary Sınıflandırma)

Öğrenci Ad Soyad:	Fatih AYIBASAN
Öğrenci No:	220707070
Bölüm / Sınıf:	Bilgisayar Mühendisliği 4. sınıf
Ders:	Yapay Zekaya Giriş
Tarih:	01.12.2025

1. Amaç ve Kapsam

Bu projenin amacı, Yapay Zekaya Giriş dersi kapsamında verilen veri setleri ile derin öğrenme/transfer öğrenme yöntemleri kullanarak tahmin (sınıflandırma) yapmak; en az 10 farklı model denemek; yeni ve hibrit modeller geliştirmek; sonuçları grafikler ve metriklerle raporlamaktır.

Rapor; veri setinin kısa tanımı ve EDA bulguları, kullanılan yöntemlerin açıklaması, her model için performans grafikleri ve metrikler, tüm modelleri tek tabloda karşılaştırma, en iyi modelin kaydedilmesi ve örnek tahmin akışını içerir.

2. Veri Setleri

2.1. Eğitim Veri Seti (Ana Kaynak)

Mendeley Data: Brain Tumor MRI Dataset (Tumor / No Tumor).

Bağlantı: https://data.mendeley.com/datasets/c9rt8d6zrf/1

2.2. Test Veri Setleri (Dış Kaynak / External Test)

1) Kaggle: Brain MRI Images for Brain Tumor Detection

Bağlantı: https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection

2) Kaggle: Brain Tumor Detection

Bağlantı: https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection

2.3. Veri Setinin Çok Kısa Tanımı

Bu çalışma, beyin MRI görüntülerinden ikili sınıflandırma yapmayı hedefler. Görüntüler iki etikete ayrılmıştır: Tumor ve No Tumor. Amaç, bir MRI görüntüsünde tümör olup olmadığını otomatik olarak tahmin etmektir.

2.4. Split ve Veri Sızıntısı (Leakage) Problemi

Orijinal veri setinde eğitim ve test ayrımı bulunsa da, EDA sırasında aynı kişiye ait birden fazla MRI görüntüsünün farklı setlere (train/test) dağılabildiği gözlemlenmiştir. Bu durum, modelin eğitimde gördüğü kişiye çok benzeyen örneklerle testte değerlendirilmesine neden olur ve metrikleri yapay şekilde yükseltir (subject leakage).

Bu nedenle, Mendeley kaynağındaki tüm görüntüler eğitim havuzu olarak kullanılmış; test değerlendirmesi, farklı kaynaklardan alınan iki bağımsız veri seti üzerinde yapılmıştır. (Mete hocanın haberi var.) Bu yaklaşım, genelleme performansını daha güvenilir ölçmeyi amaçlar.

Not: Çalışma sonunda yaklaşık ~10.000+ eğitim görüntüsü ve ~3.500 test görüntüsü olacak şekilde veri havuzları oluşturulmuştur (nihai sayılar EDA bölümünde raporlandı).

3. Keşifsel Veri Analizi (EDA)

Bu projede EDA'nın kritik çıktısı, hasta/kişi bazlı tekrarların train-test ayrımını bozabildiği tespit edildi.

3.1. Veri Envanteri

- Train havuzu: 10.085
- Train sınıf başına dağılım: Tumor: 5.019, No Tumor: 5.066
- Test havuzu: 3.253
- Test sınıf başına dağılım: Tumor: 1.655, No Tumor: 1.598

3.2. Görüntü Boyutları ve Format

Görüntülerin piksel boyutları incelendiğinde, ana eğitim kaynağında örneklerin çoğunlukla 256×256 veya 512×512 boyutlarında olduğu gözlemlenmiştir. Model eğitiminde sabit giriş boyutu gereksinimi nedeniyle tüm görüntüler belirlenen input_size değerine yeniden ölçeklendirilmiştir.

3.3. Kalite Kontrolleri

Bozuk/okunamayan, aşırı karanlık/parlak görseller bulunamamıştır.

3.4. Yakın Kopya / Subject Leakage Analizi

Aynı kişiye ait birden fazla çekimin farklı setlere dağılması, modelin gerçek genelleme yerine kişiye özgü ipuçlarını öğrenmesine yol açabiliceği tespit edilmiştir. Bu nedenle subject leakage riski EDA kapsamında analiz edilmiş ve dış kaynak test ile değerlendirme tasarımına geçilmiştir.

4. Yöntem

4.1. Ön İşleme (Preprocessing)

Bu çalışmada görüntüler klasör tabanlı olarak yüklenmiştir. Veri dizin yapısı Dataset/Training ve Dataset/Testing alt klasörleri olacak şekilde düzenlenmiş; sınıf isimleri no_tumor ve tumor olarak tanımlanmıştır. Etiket sırası sabitlenmiştir: **0 = no_tumor**, **1 = tumor**. Eğitim–doğrulama ayrımı, Training klasörü içinde **stratified** (sınıf oranlarını koruyan) biçimde yapılmıştır. Bu amaçla StratifiedShuffleSplit kullanılarak verinin **%10'u doğrulama (val)** olarak ayrılmıştır (val_ratio =

0.1, random_state = 42). Test verisi ise Testing klasöründen ayrı bir veri kümesi olarak değerlendirilmiştir.

Görüntü boyutları veri kümesinde değişken olduğu için, modele giriş boyutu sabitlenmiştir. Bu projede giriş boyutu **256×256** olarak seçilmiş ve tüm görüntüler bu boyuta yeniden ölçeklendirilmiştir. Yeniden boyutlandırma öncesinde görüntülerin en–boy oranını bozup gereksiz kırpmaya yol açmamak için **SquarePad** uygulanmıştır: görüntü kısa kenardan siyah (0) piksellerle doldurularak kare hale getirilir, ardından Resize(256,256) yapılır.

Görüntüler tensöre çevrildikten sonra normalizasyon uygulanmıştır. Normalizasyon parametreleri (mean/std) sabit elle verilmemiş; kullanılan backbone'un (timm model konfigürasyonu) varsayılan değerlerinden otomatik çekilmiştir. Seçilen özgün model custom_msaf_effb0 olduğu için mean/std değerleri **EfficientNet-B0** konfigürasyonundan alınmıştır (resolve_data_config). Bu sayede önceden eğitilmiş (pretrained) modellerin beklediği ölçeklendirme ile uyum sağlanmıştır.

Tekrar üretilebilirlik için rastgelelik kontrol altına alınmıştır: Python/NumPy/PyTorch seed ayarlanmış, DataLoader işçileri için seed_worker kullanılmıştır. Eğitim GPU varsa CUDA üzerinde yürütülmüş, yoksa CPU kullanılmıştır.

4.2. Veri Artırma (Augmentation)

Bu çalışmada veri artırma yalnızca **eğitim (train)** aşamasında uygulanmış; doğrulama (val) ve test aşamalarında artırma kapatılmıştır. Artırma şiddeti tek bir parametre ile kontrol edilmiştir: aug_strength. Bu parametre 0.0 olduğunda eğitim dönüşümleri doğrulama dönüşümleri ile aynı hale gelir ve augmentasyon **tamamen devre dışı** kalır. aug_strength arttıkça augmentasyonun olasılıkları ve büyüklükleri kademeli olarak artacak şekilde tasarlanmıştır. Şiddet ölçeklemesi için parametre 0–1 aralığına taşınmış ve tüm artırmalar bu ölçek ile çarpılmıştır (şiddet arttıkça flip/rotasyon/perspektif vb. dönüşümlerin hem uygulanma ihtimali hem de etkisi artar).

Uygulanan artırmalar, tensöre çevirmeden önce (PIL seviyesinde) aşağıdaki dönüşümlerden oluşur:

- RandomHorizontalFlip (yatay çevirme)
- RandomVerticalFlip (dikey çevirme)
- RandomRotation (dönme)
- RandomAffine (çeviri + ölçekleme)
- RandomPerspective (perspektif bozulması)
- ColorJitter (parlaklık/kontrast/doygunluk/hue)

- GaussianBlur (bulanıklaştırma)

Ayrıca tensör seviyesinde, düzenlileştirme (regularization) amaçlı **RandomErasing** uygulanmıştır. Bu yöntem, görüntü üzerinde rastgele küçük bölgeleri silerek modelin tek bir lokal bölgeye aşırı bağımlı olmasını azaltmayı hedefler.

Augmentation şiddeti için yapılan deneyler: Bu projede aug_strength değeri sistematik şekilde denenmiştir. Önce **0.0–0.3** aralığında denemeler yapılmış ve **0.3 değerinde yaklaşık %90 doğruluk** elde edilmiştir. Daha sonra **0.3–0.7** aralığında her değer tek tek denenerek eğitim tekrarlanmış; bu aralıktaki tüm denemelerde doğruluk **yaklaşık %90 seviyesinde kalmış**, yani doğrulukta anlamlı bir artış gözlenmemiştir. Buna karşılık daha düşük şiddetlerde performans düşmüştür: aug_strength = 0.2 denendiğinde doğruluk **%89** seviyesine, aug_strength = 0.1 denendiğinde ise **%85** seviyesine gerilemiştir. Bu sonuç, augmentasyonun belirli bir minimum şiddetin altında modele yeterli genelleme katkısı yapmadığını; 0.3 ve üzerindeki aralıkta ise doğruluk açısından **doygunluğa (plateau)** ulaşıldığını göstermektedir.

Bu nedenle nihai model eğitiminde augmentasyon, genelleme katkısı ile stabiliteyi birlikte sağlayan şiddet aralığında (özellikle **0.3 ve üzeri**) tercih edilmiştir.

4.3. Overfitting'i Engelleme Stratejileri

Overfitting'i azaltmak için birden fazla mekanizma birlikte kullanılmıştır:

1. **Pretrained backbone + kontrollü baş (head):** Özgün modelde EfficientNet-B0 backbone pretrained olarak kullanılmış, sınıflandırma için ayrı bir MLP head tasarlanmıştır.
2. **Düzenlileştirme (regularization):**

- Head katmanlarında **Dropout** kullanılmıştır (özgün modelde dropout = 0.35).
- Veri artırma (train-only) uygulanmıştır.
- Optimizasyonda **AdamW + weight decay** kullanılarak L2 benzeri düzenlileştirme sağlanmıştır.

3. **Early Stopping:** Eğitim sırasında doğrulama kaybı (val_loss) izlenmiş, en iyi val_loss görüldüğünde model ağırlıkları kaydedilmiş; val_loss iyileşmezse sayaç artırılarak **patience = 7** sonunda eğitim durdurulmuştur. Böylece gereksiz epoch'larda ezberleme azaltılmıştır.
4. **LR Scheduler:** ReduceLROnPlateau kullanılmıştır (mode = "min", factor = 0.5, patience = 3). Val_loss plato yaptığında öğrenme oranı düşürülerek daha stabil yakınsama hedeflenmiştir.

5. **Sınıf dengesizliği (class imbalance) yönetimi:** Eğitim setinde sınıf sayıları farklı olabileceği için compute_class_weights ile ters frekans bazlı sınıf ağırlıkları hesaplanmış ve kayıp fonksiyonunda kullanılmıştır (CrossEntropyLoss(weight=class_weights)).

4.4. Parametre Optimizasyonu

Hiperparametre seçimi için **Random Search** uygulanmıştır. Arama uzayı şu parametreleri içerir:

- Öğrenme oranı (lr): log-uniform dağılım, **1e-5 – 3e-3**
- Weight decay (weight_decay): log-uniform dağılım, **1e-6 – 1e-3**
- Batch size (batch_size): ayrık seçim, **[8, 16, 24, 32]**

Random search ayarları:

- Deneme sayısı: **N_TRIALS = 15**
- Her deneme için epoch sayısı: **MAX_EPOCHS = 10**
- Seçim metriği: **validation accuracy (val_acc)**, büyük olan daha iyi kabul edilmiştir.

Her deneme, deterministikliği korumak amacıyla farklı ama kontrol edilen seed ile çalıştırılmıştır (seed = 42 + i). Random search sonucunda en iyi konfigürasyon seçilmiş ve nihai eğitim bu değerlerle daha uzun eğitim süresinde yapılmıştır.

4.5. Değerlendirme Metrikleri

Tüm modeller için Accuracy ve Loss eğrileri; Confusion Matrix; ROC eğrisi ve AUC; ayrıca tablo halinde precision, recall (sensitivity), F1-score ve Cohen’s kappa raporlanmıştır:

Eğitim sürecinde her epoch sonunda:

- **Train loss / Train accuracy**
- **Validation loss / Validation accuracy**
  hesaplanmış ve kaydedilmiştir. Bu değerler kullanılarak Accuracy ve Loss eğrileri oluşturulmuştur.

Final değerlendirmede sınıflandırma performansı aşağıdaki metriklerle raporlanmıştır:

- **Confusion Matrix**
- **Accuracy**

- **Precision**
- **Recall / Sensitivity**
- **F1-score**
- **Cohen's Kappa**
- **ROC eğrisi ve AUC**

Model çıktısı iki sınıf logitidir. Tümör olasılığı, softmax sonrası **pozitif sınıf (tumor=1)** olasılığı olarak alınmıştır. Sınıf kararı için sabit 0.5 eşiğine körü körüne bağlı kalınmamış; önce **yalnızca validation set** üzerinde 0–1 aralığında ince tarama yapılmıştır (1001 adet eşik) ve **F1-score'u maksimize eden threshold** seçilmiştir. Bu eşik seçimi test verisine dokunmadan yapılmış, daha sonra aynı threshold ile test seti değerlendirilmiştir. Böylece test seti "tuning" için kullanılmadığından değerlendirme daha adil kalır.

Çıktı olarak her model koşusunda şu dosyalar üretilmiştir:

- accuracy.png, loss.png (eğitim eğrileri)
- confusion_matrix.png
- roc_curve.png
- metrics_table.png (precision/recall/f1/kappa/accuracy)
- best_threshold.txt
- best.pt (model ağırlıkları)

4.6. Çalışma Akışı (Flowchart)

Bu projede uçtan uca işlem hattı aşağıdaki adımlardan oluşur. Flowchart'ta her madde ayrı kutu olacak şekilde gösterilmiştir:

1. **Veri kaynaklarının hazırlanması ve ayrımı**
    - Ana eğitim havuzu: Mendeley veri seti (tumor/no_tumor).
    - Dış kaynak test: iki ayrı Kaggle veri seti.
      (Amaç: aynı kişiye ait tekrar görüntülerden kaynaklı subject leakage riskini azaltmak ve genellemeyi dış testte ölçmek.)
2. **Çalışma ortamının kurulması ve tekrar üretilebilirlik**

- CPU/GPU seçimi, seed ayarlama (Python/NumPy/PyTorch), DataLoader worker seed kontrolü.

3. **Transform / preprocessing pipeline'ının tanımlanması**
   - SquarePad → Resize (sabit input_size) → ToTensor → Normalize
   - Train için opsiyonel: Augmentation (aug_strength'e bağlı) + RandomErasing
   - Val/Test: augmentasyon kapalı, sadece preprocess.
4. **Eğitim–doğrulama ayrımı (stratified split)**
   - Training havuzundan sınıf oranlarını koruyacak şekilde validation ayrılır.
5. **DataLoader oluşturma**
   - Train / Val / Test için batch, shuffle ve num_workers ayarları ile yükleyicilerin hazırlanması.
6. **Model kurulumunun yapılması ve 10 modelin denenmesi**
   Bu projede toplam **10 farklı model** eğitilmiştir:
   - **7 adet normal (tek-backbone) model:** Standart transfer öğrenme yaklaşımıyla tek bir backbone üzerinden sınıflandırma.
   - **2 adet hibrit model:** İki farklı backbone'un özelliklerini birleştiren mimariler.
     - **Hibrit-1: DenseNet121 + EfficientNetB0**
     - **Hibrit-2: EfficientNetB0 + Swin-T**
   - **1 adet özgün (kendi geliştirdiğimiz) model: Multi-Scale feature extraction + attention/fusion** yaklaşımı içeren özel mimari (multiscale feature çıkarımı yapılan model).

Flowchart'ta bu adım, "Model Seçimi / Model Profili" kutusu altında **(Normal / Hibrit / Custom)** olarak üç dala ayrılacak şekilde gösterilmiştir.

7. **Kayıp fonksiyonu ve optimizasyon ayarları**
   - (Varsa) class weight ile ağırlıklı loss, optimizer (AdamW), weight decay, LR scheduler.
8. **Eğitim döngüsü**
   - Epoch bazlı train→val ilerleyişi, metriklerin her epoch sonunda kaydı (loss/accuracy).

- (Varsa) mixed precision (AMP), gradient scaling.

9. **Early Stopping ve en iyi ağırlıkların seçimi**
    - Validation performansına göre en iyi model checkpoint'inin kaydedilmesi ve aşırı öğrenmenin durdurulması.
10. **Threshold seçimi (sadece validation üzerinde)**

- Pozitif sınıf olasılığı için farklı eşikler denenir, seçilen kriter (örn. F1 maksimum) ile en iyi threshold belirlenir.

11. **Dış test değerlendirmesi (external test)**

- Seçilen threshold ile test setinde Confusion Matrix, ROC/AUC ve diğer metrikler hesaplanır.

12. **Sonuçların kaydedilmesi ve raporlanması**

- Accuracy/Loss eğrileri, Confusion Matrix, ROC eğrisi, metrics table görselleri ve özet karşılaştırma tablosu üretilir.
- En iyi model ayrıca kaydedilir ve örnek tahmin akışı rapora eklenir.

Flowchart çizimi için net şema (kutular böyle olsun)

- **Data Sources → Preprocess & Augmentation → Split (Train/Val) → Dataloader → Model Branch**
    - **Normal Models (7)**
    - **Hybrid-1: DenseNet121 + EfficientNetB0**
    - **Hybrid-2: EfficientNetB0 + Swin-T**
    - **Custom: Multi-Scale Feature + Fusion**
      → Train + Scheduler + EarlyStopping → Threshold Selection (Val) → External Test Metrics → Save Results & Best Model
    -

5. Modeller ve Sonuçlar

Bu bölümde 13 farklı model raporlanır. Her model için: (i) kısa mimari/yöntem açıklaması, (ii) eğitim ayarları, (iii) Accuracy-Loss grafikleri, (iv) Confusion Matrix, (v) ROC eğrisi ve (vi) metrik tablosu verilerek yorumlanır.

5.1. Tüm Modellerin Karşılaştırma Tablosu

Model	Accuracy	Precision	Recall/Sensitivity	F1-Score	Kappa	AUC
my_model_with_aug0.3.pt	0.9084	0.9978	0.8218	0.9013	0.8173	0.9884
hybrid_dn121_effb0_with_aug_0.3.pt	0.8607	1.0	0.7262	0.8414	0.7227	0.9665
hybrid_dn121_effb0.pt	0.8392	1.0	0.6839	0.8123	0.6801	0.9389
my_model.pt	0.8054	0.9990	0.6181	0.7637	0.6133	0.9355
hybrid_swinT_effb0_with_aug_0.3.pt	0.7949	0.9969	0.5987	0.7482	0.5926	0.9749
resnet34.pt	0.7943	0.9989	0.5963	0.7468	0.5914	0.9538
Densenet121.pt	0.7851	1.0	0.5776	0.7323	0.5733	0.9620
convnext_tiny.pt	0.7746	1.0	0.5570	0.7155	0.5527	0.9603
hybrid_swinT_effb0.pt	0.7448	1.0	0.4984	0.6653	0.4940	0.9562
resnet50.pt	0.7193	1.0	0.4483	0.6191	0.4439	0.9620
inception_v3.pt	0.7101	1.0	0.4302	0.6016	0.4258	0.9005
efficientnet_b0.pt	0.6925	0.9969	0.3969	0.5678	0.3915	0.9027
mobilenetv2_100.pt	0.6387	1.0	0.2900	0.4496	0.2864	0.8894

Tablo 1. Tüm modellerin metriklerle karşılaştırması.

5.2. Model 1: My_model_aug_0.3

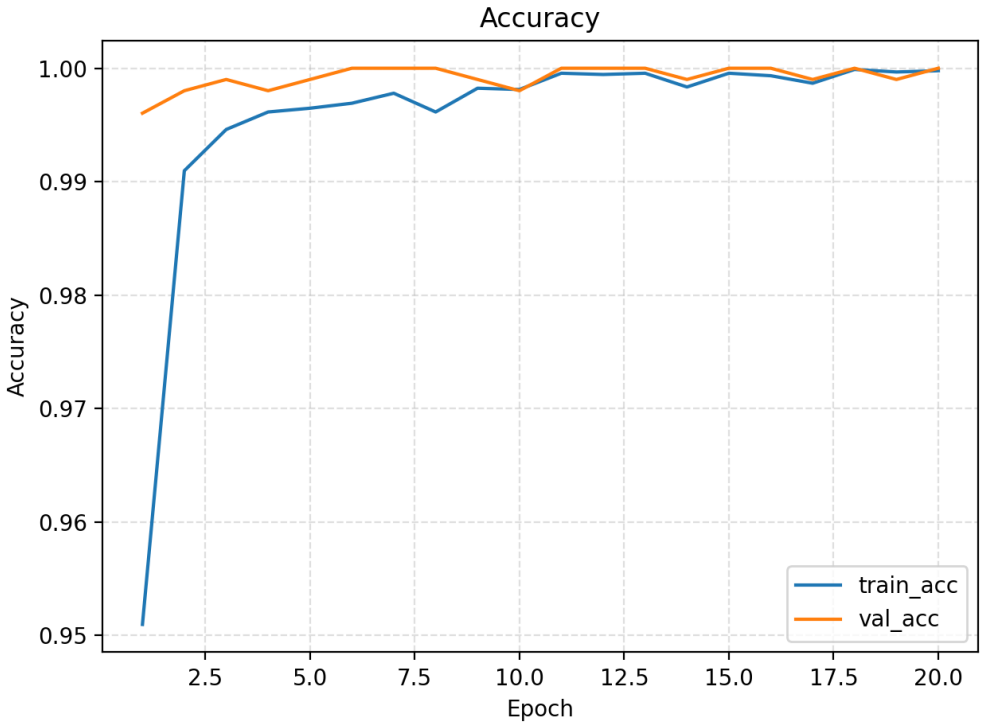
Eğitim Ayarları

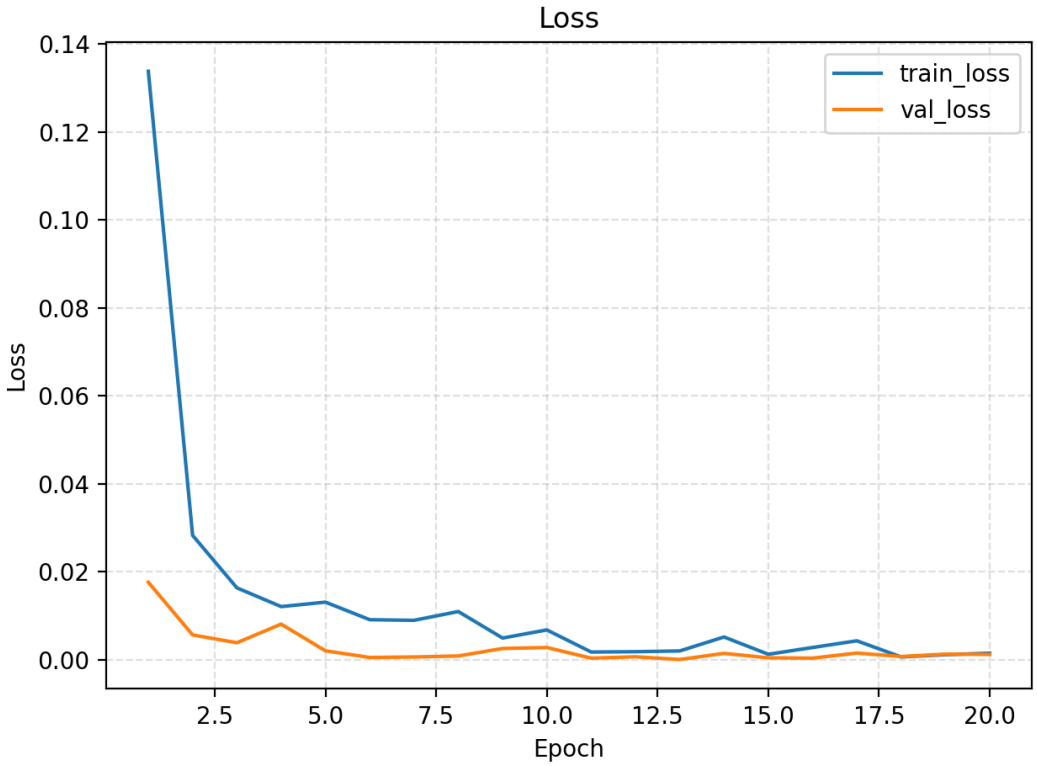
Proje klasörlerinde model ismi içerisindeki her kod dosyası o modele özgüdür random search sonuçları içerisinde kayıtlıdır.

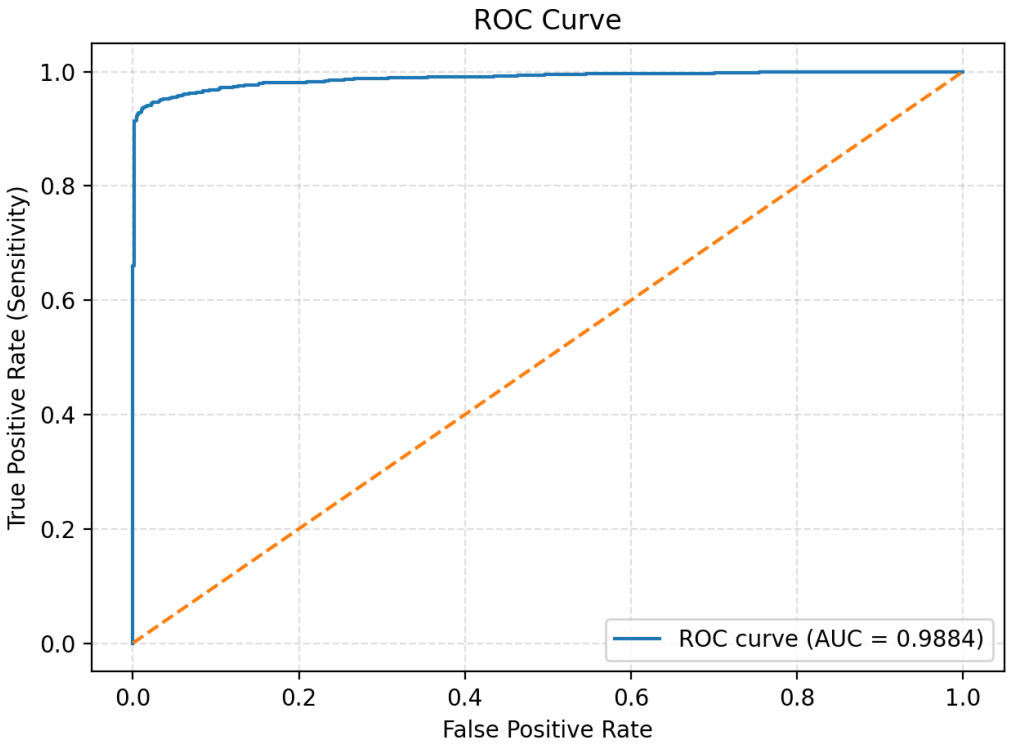
Sonuç Grafikleri

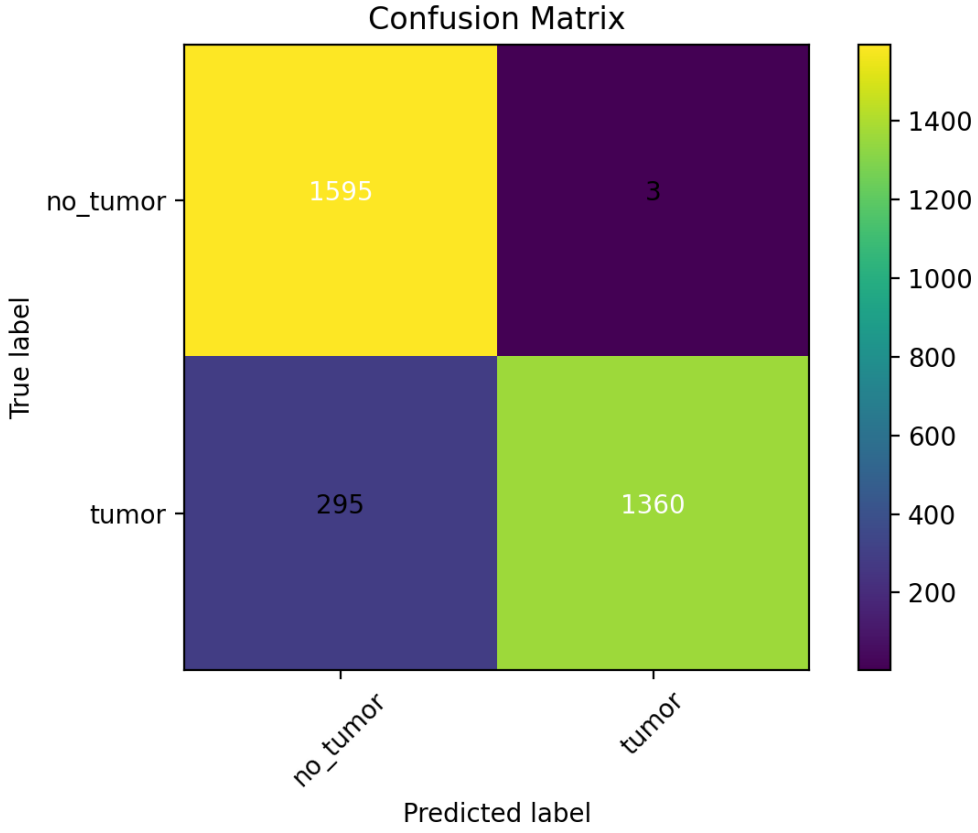
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
0.9978	0.8218	0.9013	0.8173	0.9084




Loss
train_loss
val_loss
0.14
0.12
0.10
0.08
0.06
0.04
0.02
0.00
Loss
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9884)

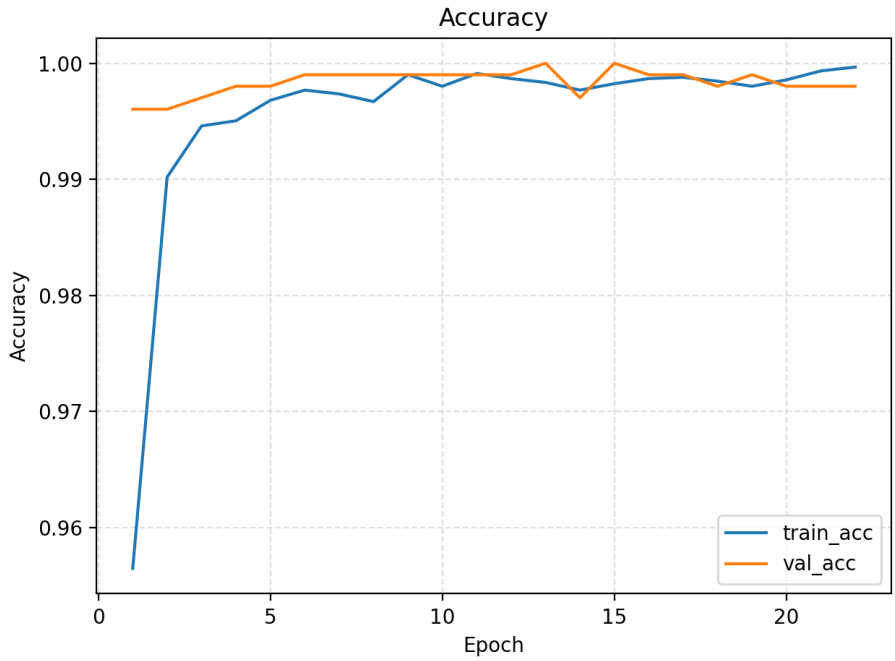

Confusion Matrix
no_tumor
tumor
True label
1595
3
295
1360
1400
1200
1000
800
600
400
200
no_tumor
tumor
Predicted label

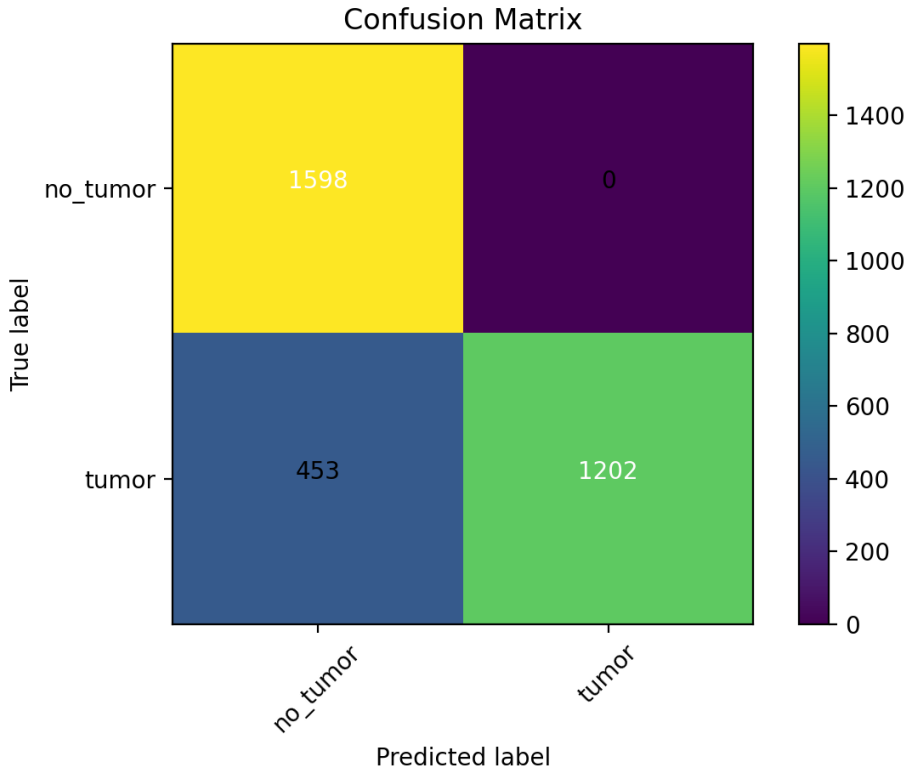
5.3. Model 2: hybrid_dn121_effb0_with_aug_0.3.pt

Eğitim Ayarları

Proje klasörlerinde model ismi içerisindeki her kod dosyası o modele özgüdür random search sonuçları içerisinde kayıtlıdır.

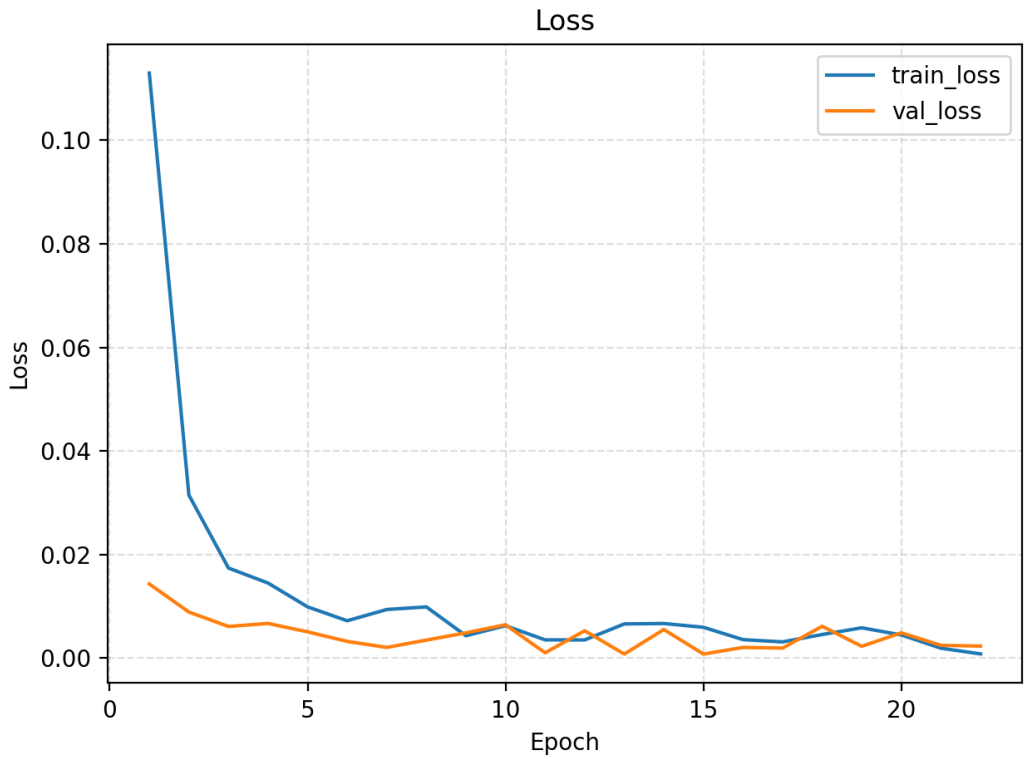
Sonuç Grafikleri

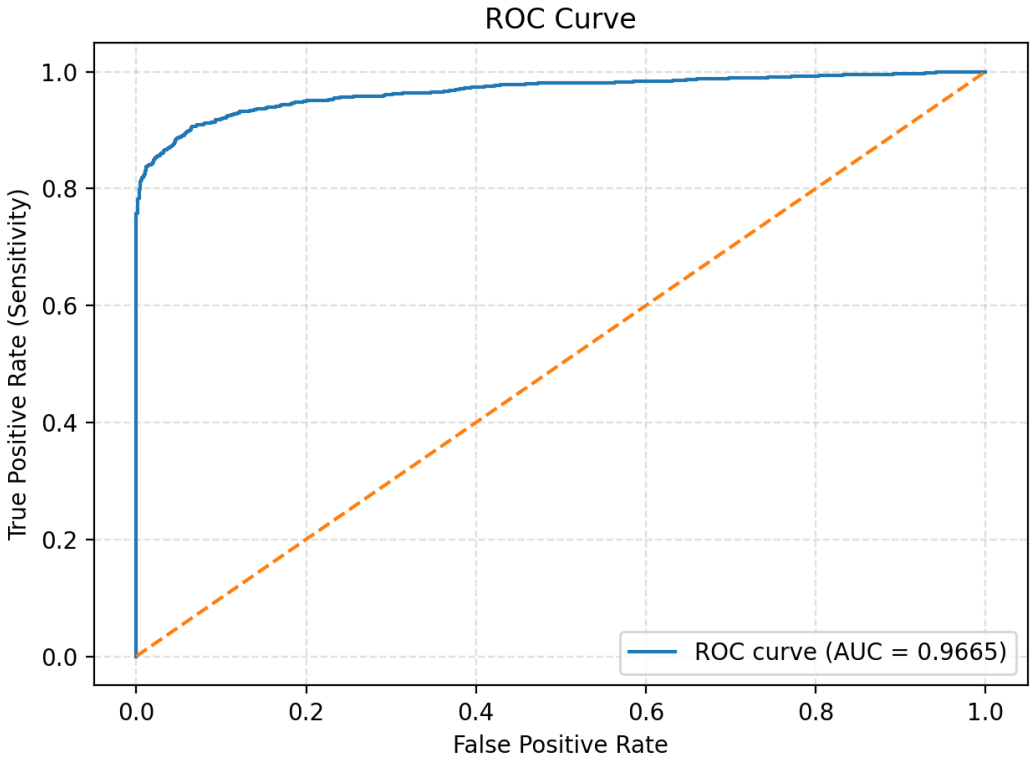

Accuracy
1.00
0.99
0.98
0.97
0.96
Accuracy
0
5
10
15
20
Epoch
train_acc
val_acc


Confusion Matrix
no_tumor
tumor
True label
1598
0
453
1202
1400
1200
1000
800
600
400
200
0
no_tumor
tumor
Predicted label

Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.7263	0.8414	0.7228	0.8607


Loss
train_loss
val_loss
Loss
Epoch
0.00
0.02
0.04
0.06
0.08
0.10
0
5
10
15
20


ROC Curve
ROC curve (AUC = 0.9665)
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0

.

5.4. Model 3: hybrid_dn121_effb0.pt

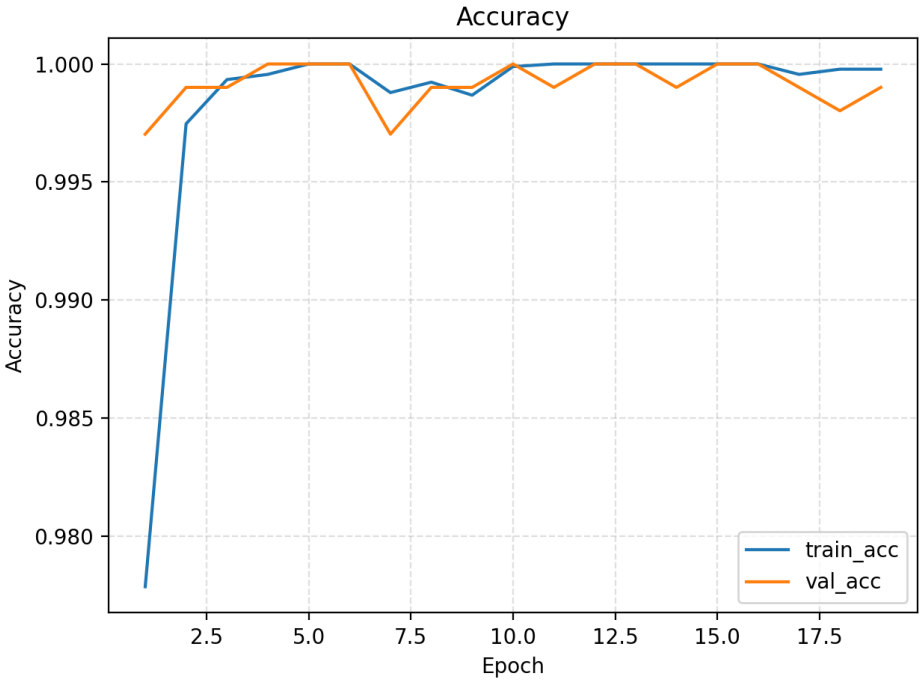
Eğitim Ayarları

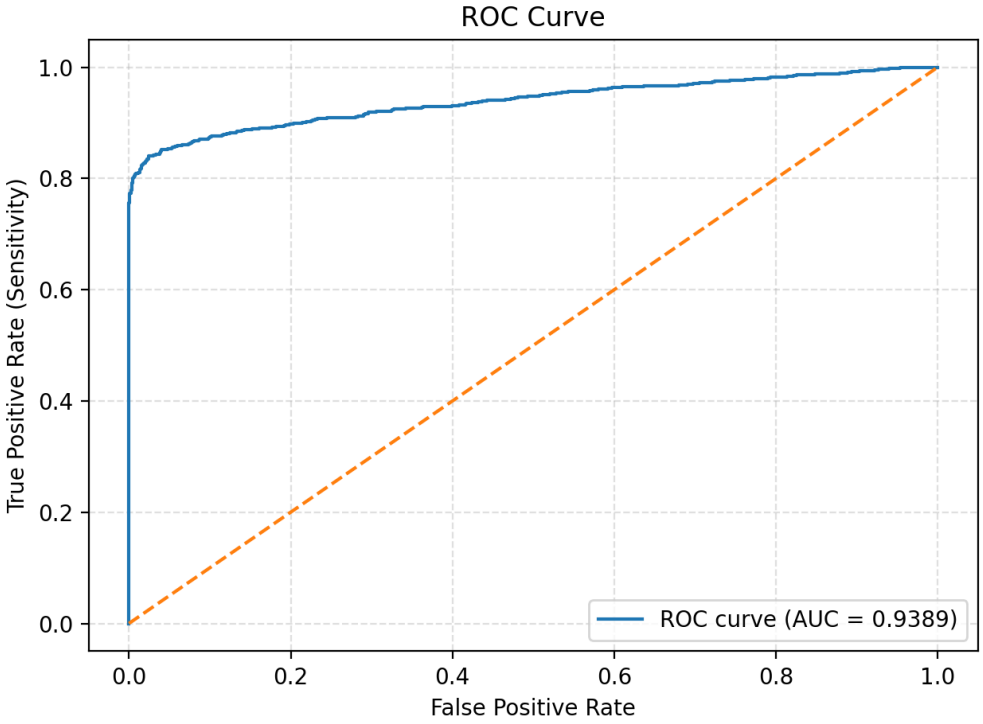
Proje klasörlerinde model ismi içerisindeki her kod dosyası o modele özgüdür random search sonuçları içerisinde kayıtlıdır.

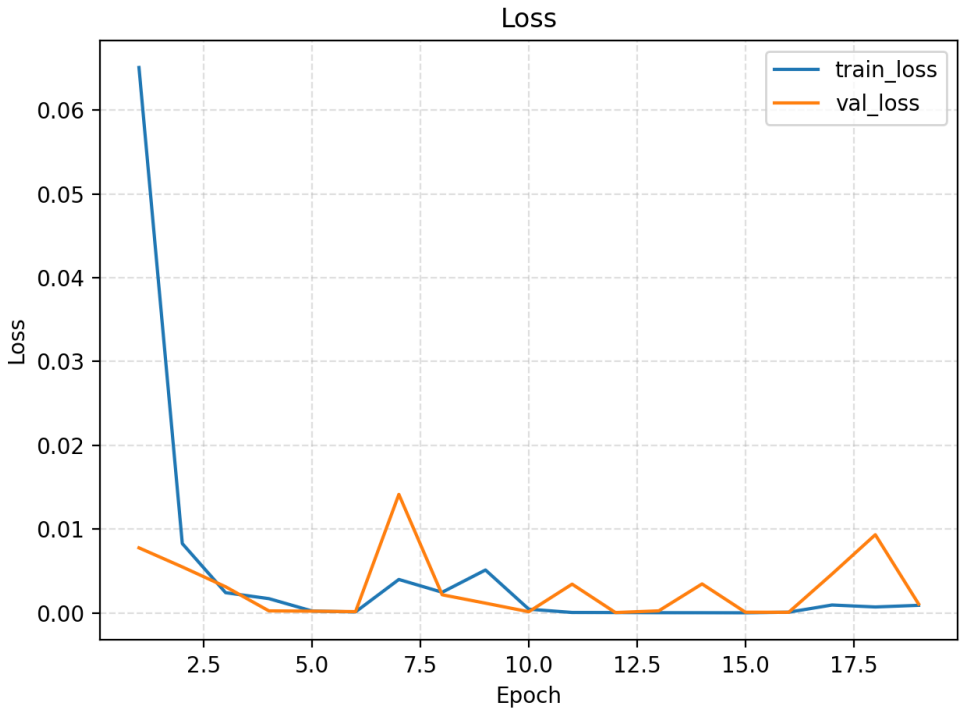
Sonuç Grafikleri

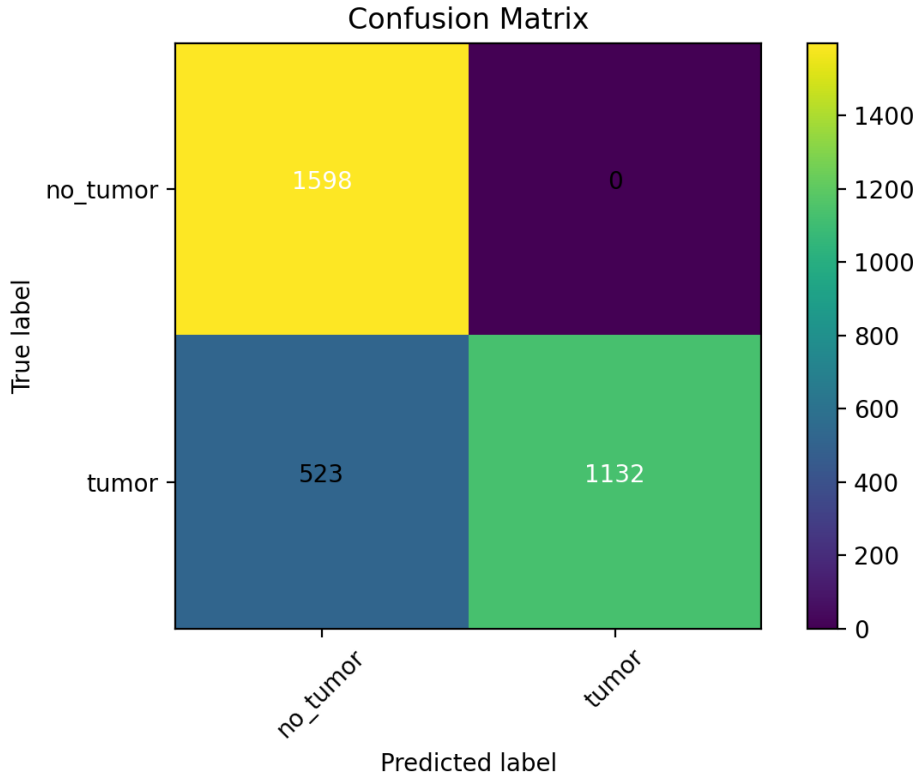
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.6840	0.8123	0.6802	0.8392


Accuracy
1.000
0.995
0.990
0.985
0.980
Accuracy
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch
train_acc
val_acc


ROC Curve
1.0
0.8
0.6
0.4
0.2
0.0
True Positive Rate (Sensitivity)
0.0
0.2
0.4
0.6
0.8
1.0
False Positive Rate
ROC curve (AUC = 0.9389)


Loss
train_loss
val_loss
Loss
Epoch
0.00
0.01
0.02
0.03
0.04
0.05
0.06
2.5
5.0
7.5
10.0
12.5
15.0
17.5

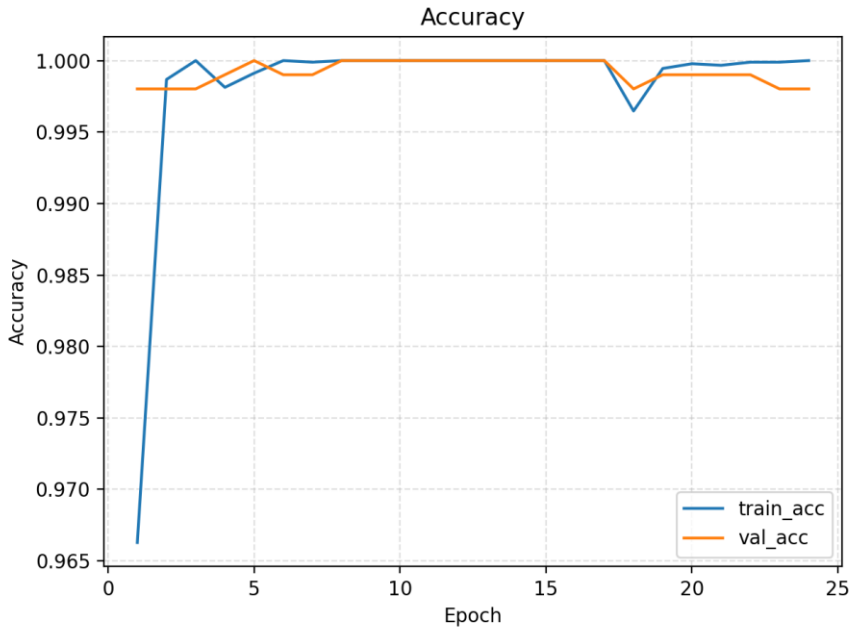

Confusion Matrix
no_tumor
tumor
True label
1598
0
523
1132
no_tumor
tumor
Predicted label
0
200
400
600
800
1000
1200
1400

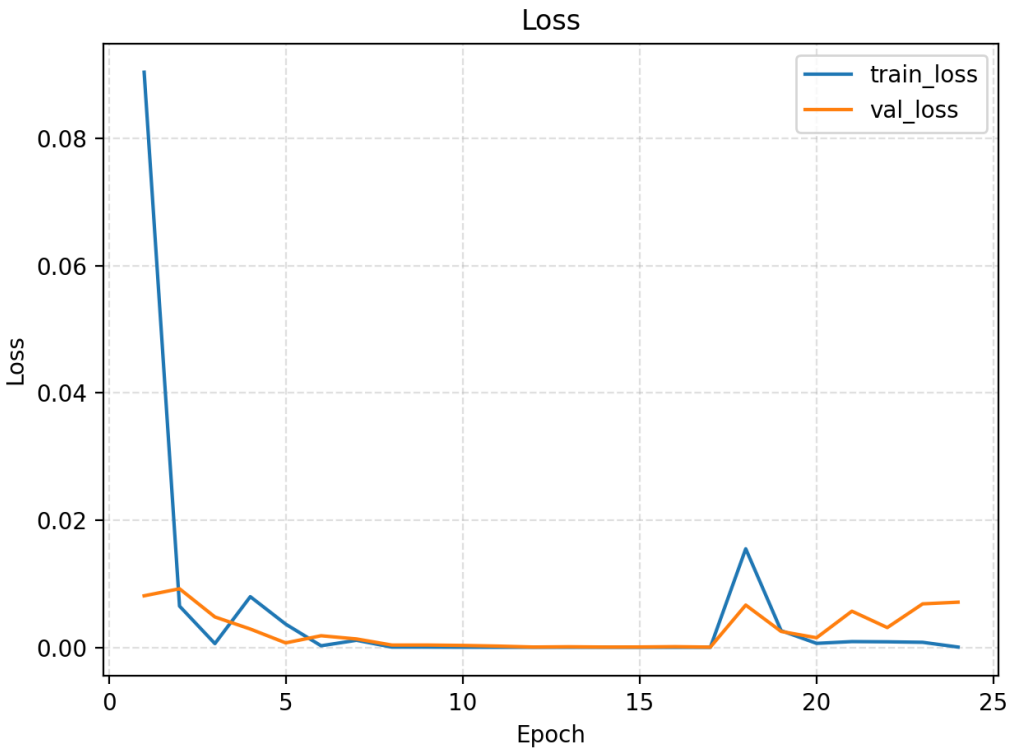
5.5. Model 4: my_model.pt

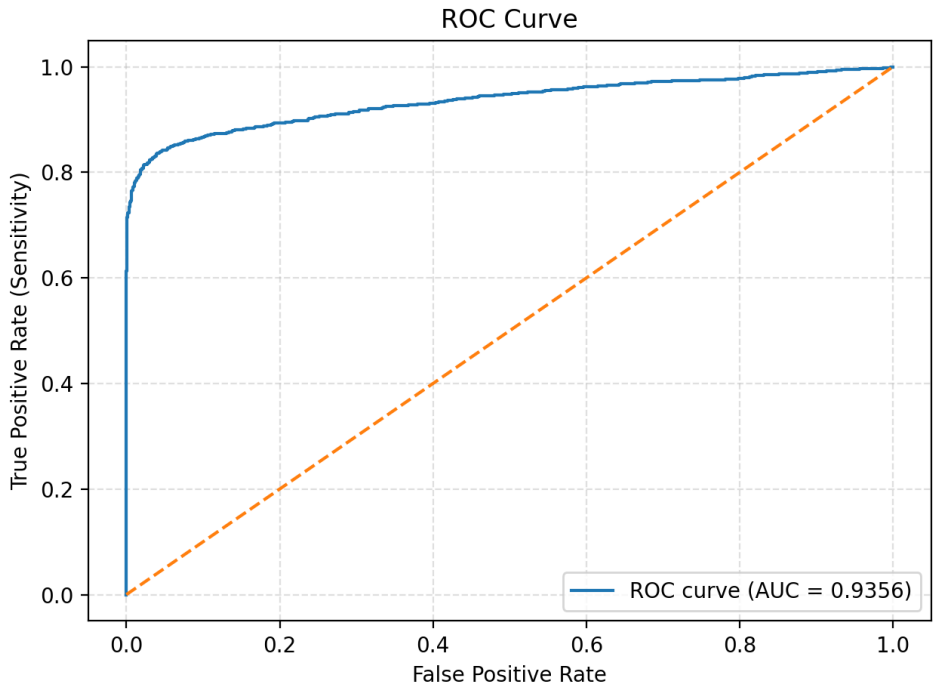
Sonuç Grafikleri

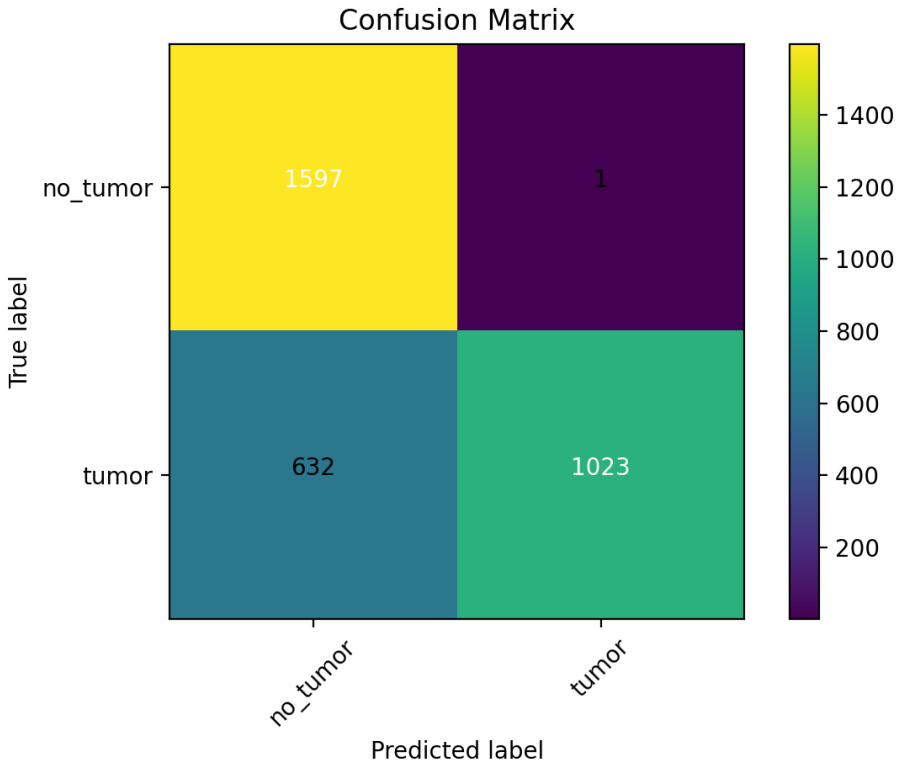
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
0.9990	0.6181	0.7637	0.6133	0.8054




Loss
train_loss
val_loss
0.08
0.06
0.04
0.02
0.00
Loss
0
5
10
15
20
25
Epoch





Yorum

Augmentasyon edildikten sonra ciddi doğruluk kazanıldığı tespit edildi.

5.6. Model 5: hybrid_swinT_effb0_with_aug_0.3.pt

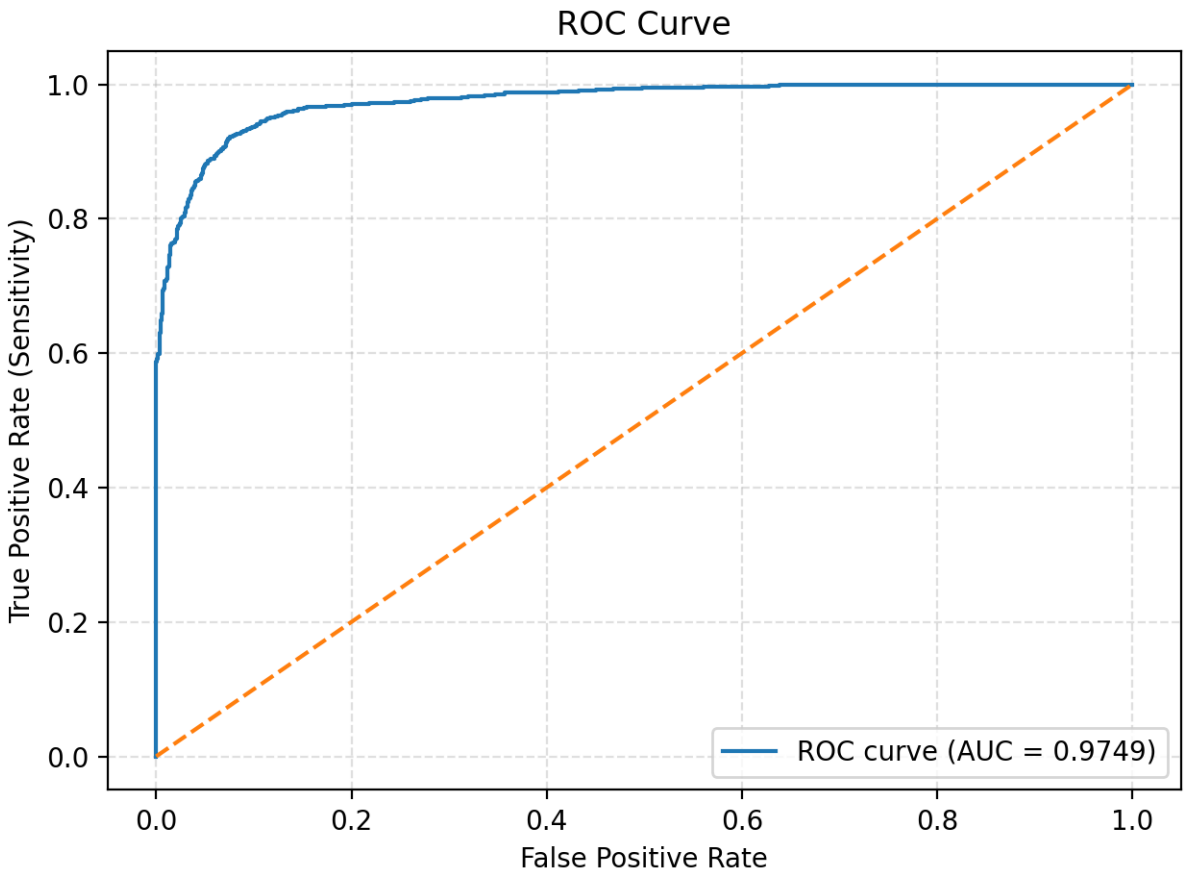
Eğitim Ayarları

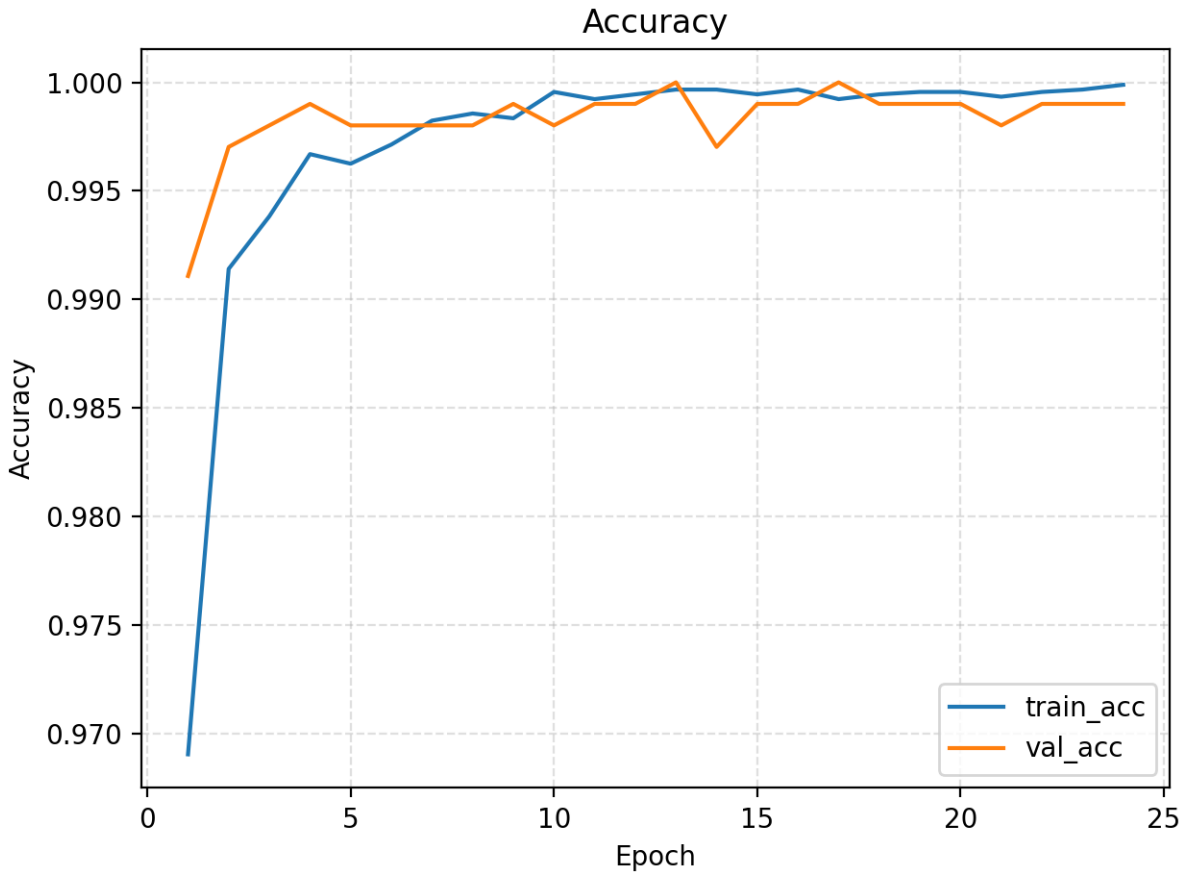
Proje klasörlerinde model ismi içerisindeki her kod dosyası o modele özgüdür random search sonuçları içerisinde kayıtlıdır.

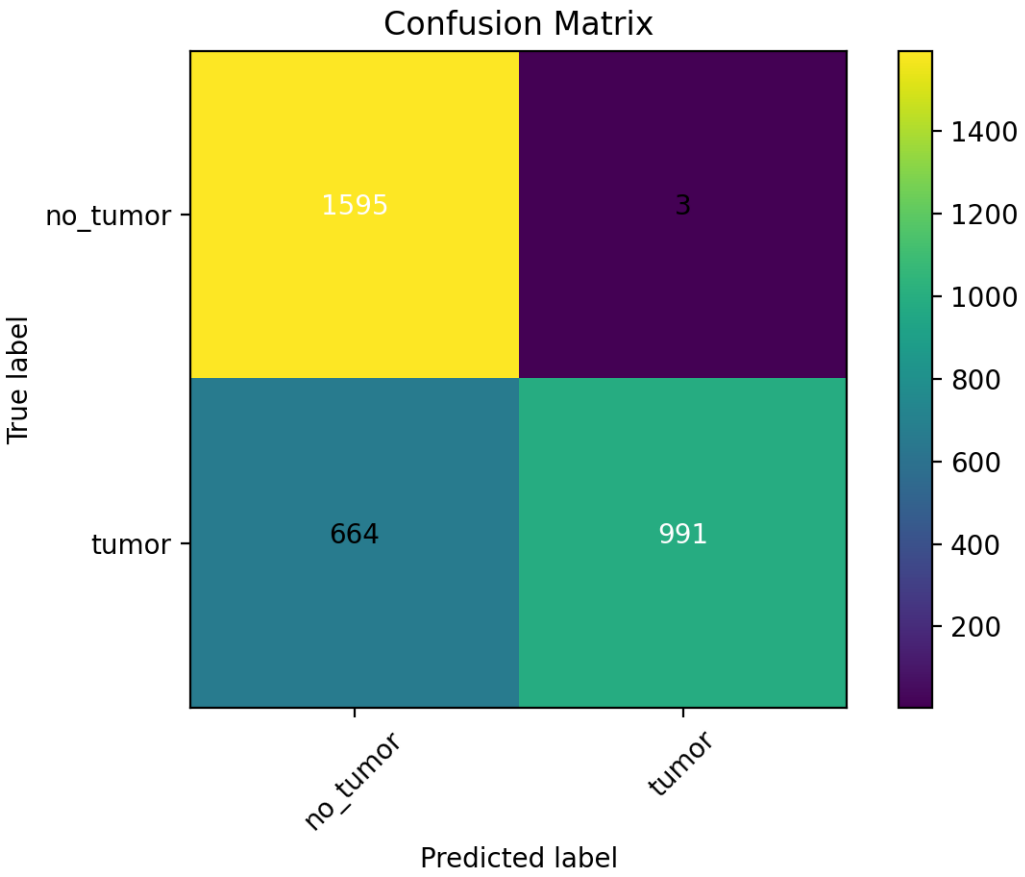
Sonuç Grafikleri

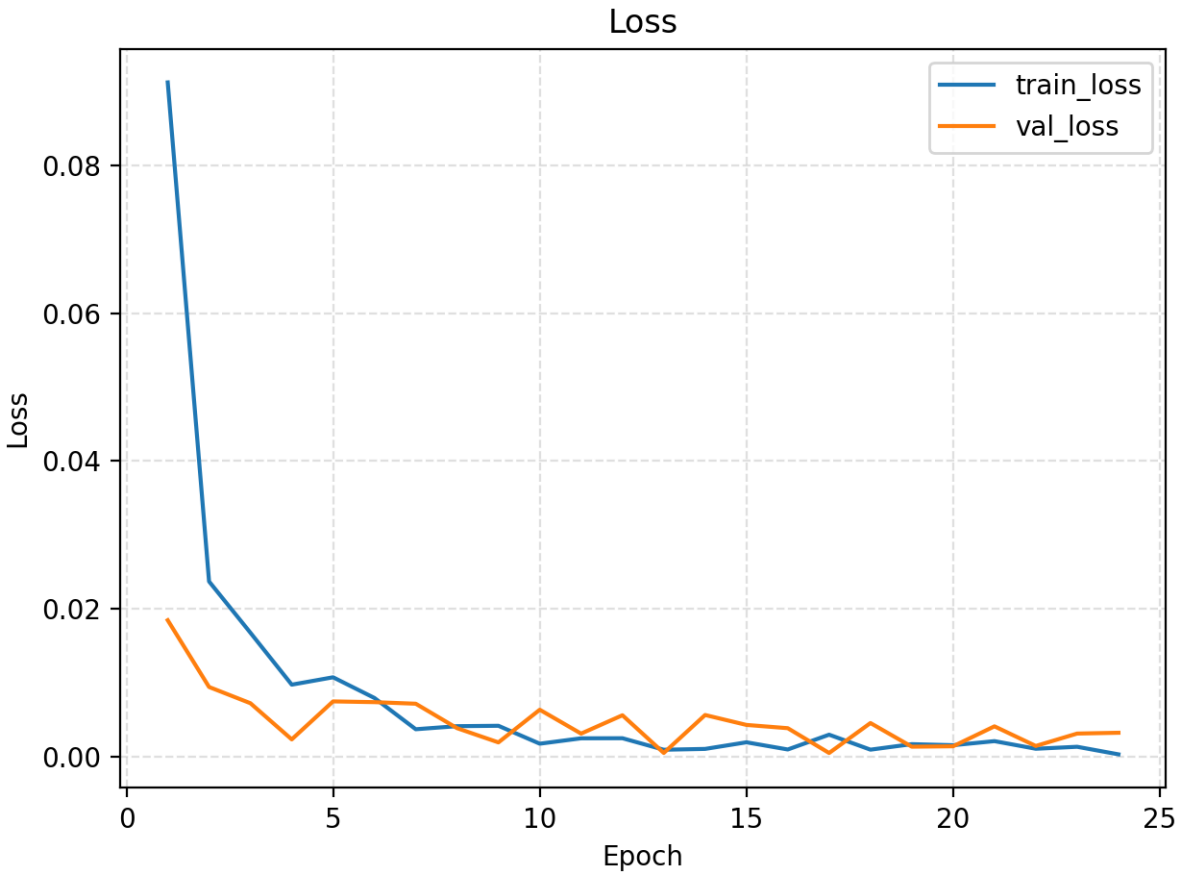
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
0.9970	0.5988	0.7482	0.5927	0.7950


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9749)


Accuracy
1.000
0.995
0.990
0.985
0.980
0.975
0.970
Accuracy
0
5
10
15
20
25
Epoch
train_acc
val_acc


Confusion Matrix
1595
3
664
991
no_tumor
tumor
True label
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200

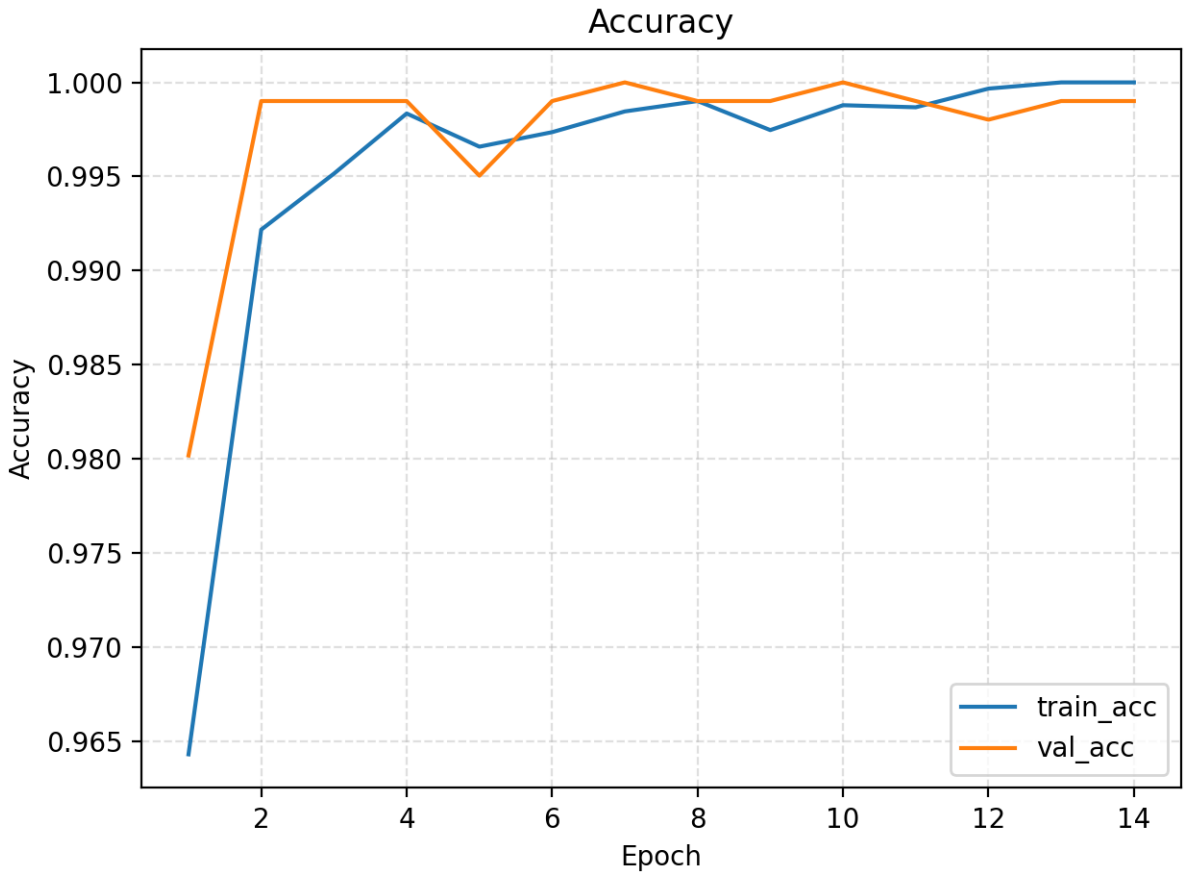


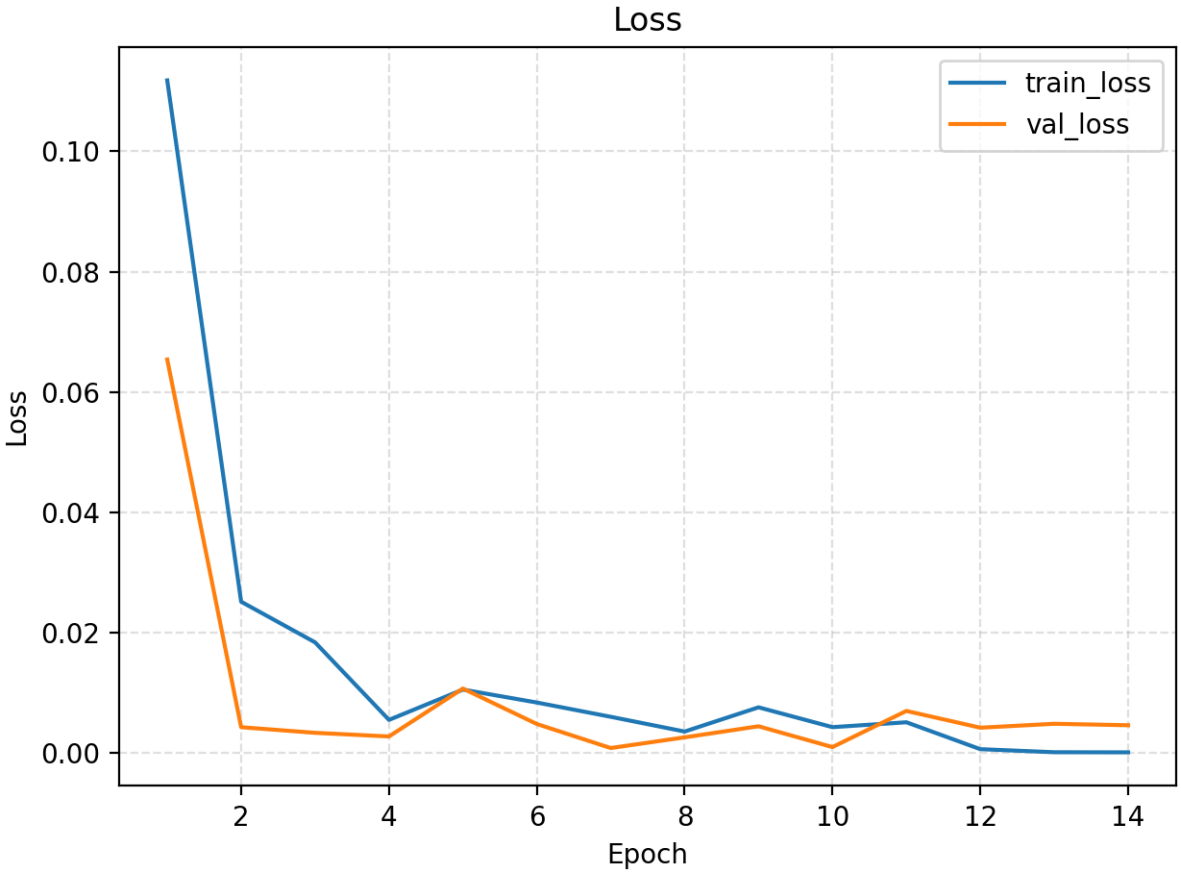
5.7. Model 6: resnet34.pt

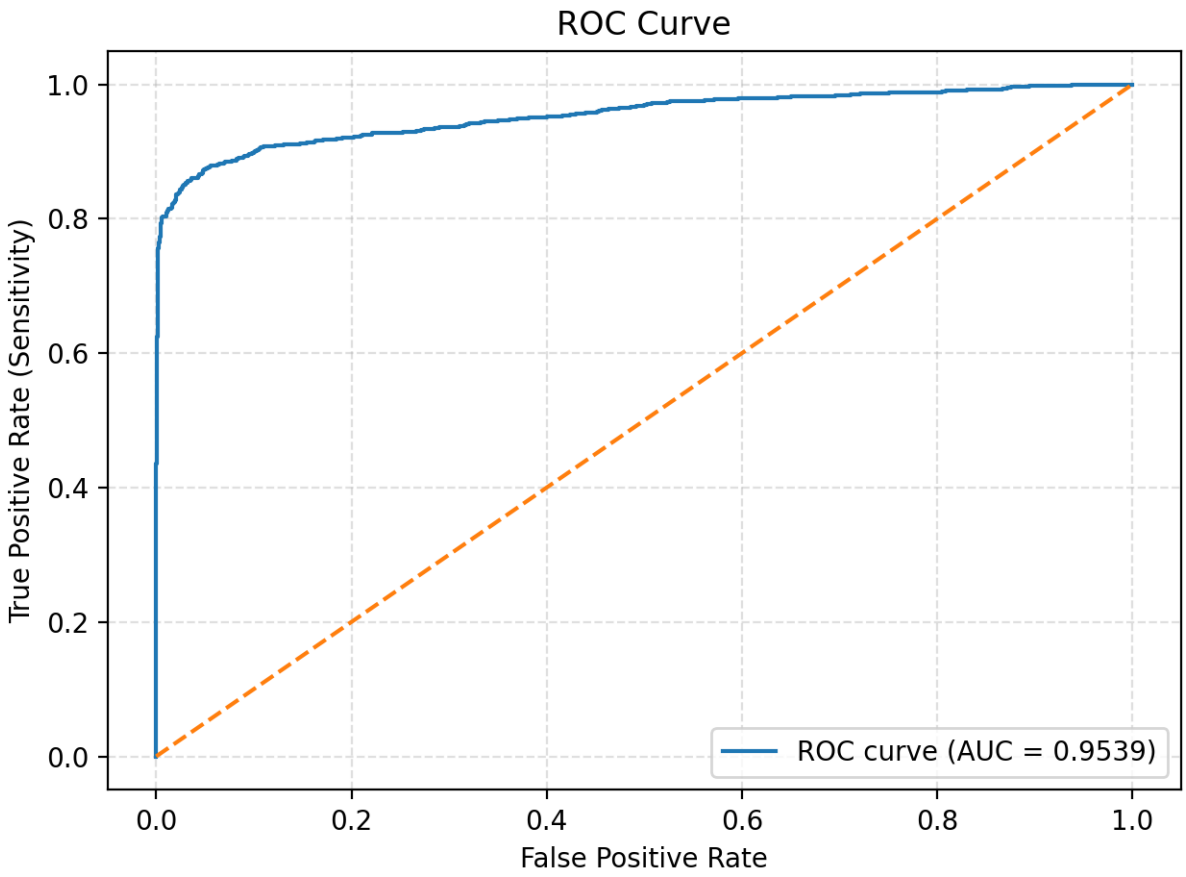
Sonuç Grafikleri

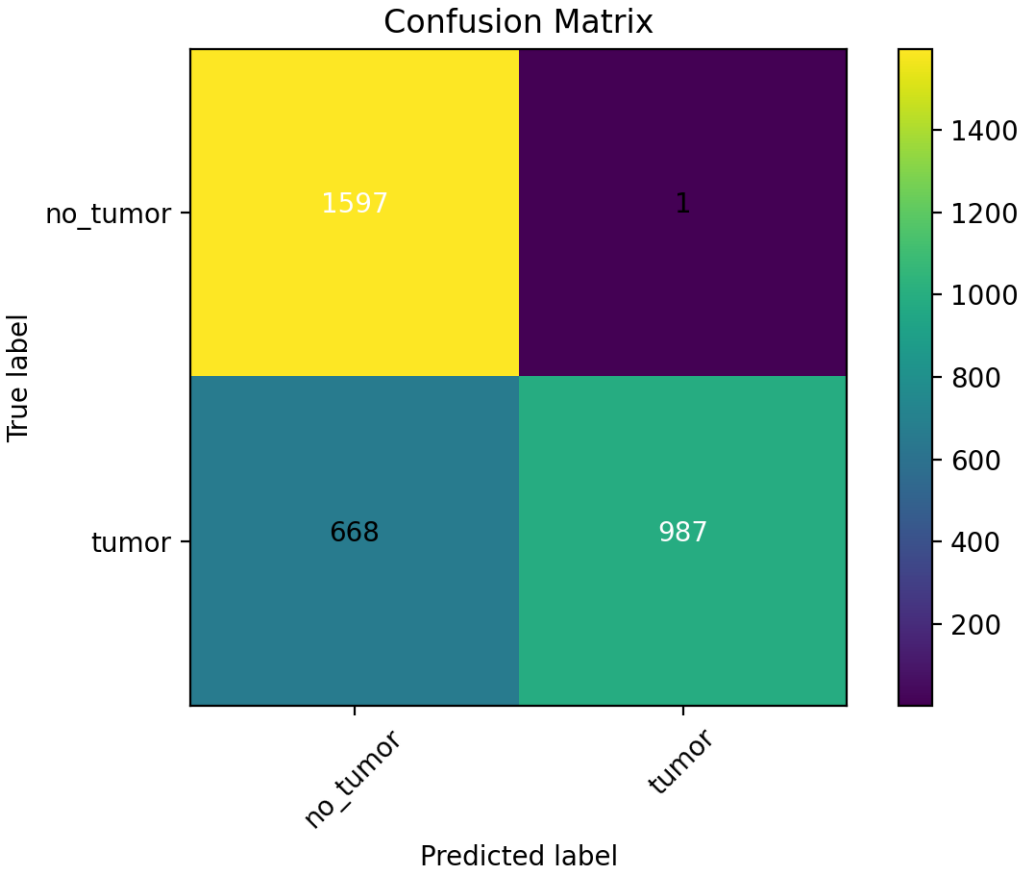
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
0.9990	0.5964	0.7469	0.5915	0.7943




Loss
train_loss
val_loss
0.10
0.08
0.06
0.04
0.02
0.00
Loss
2
4
6
8
10
12
14
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9539)

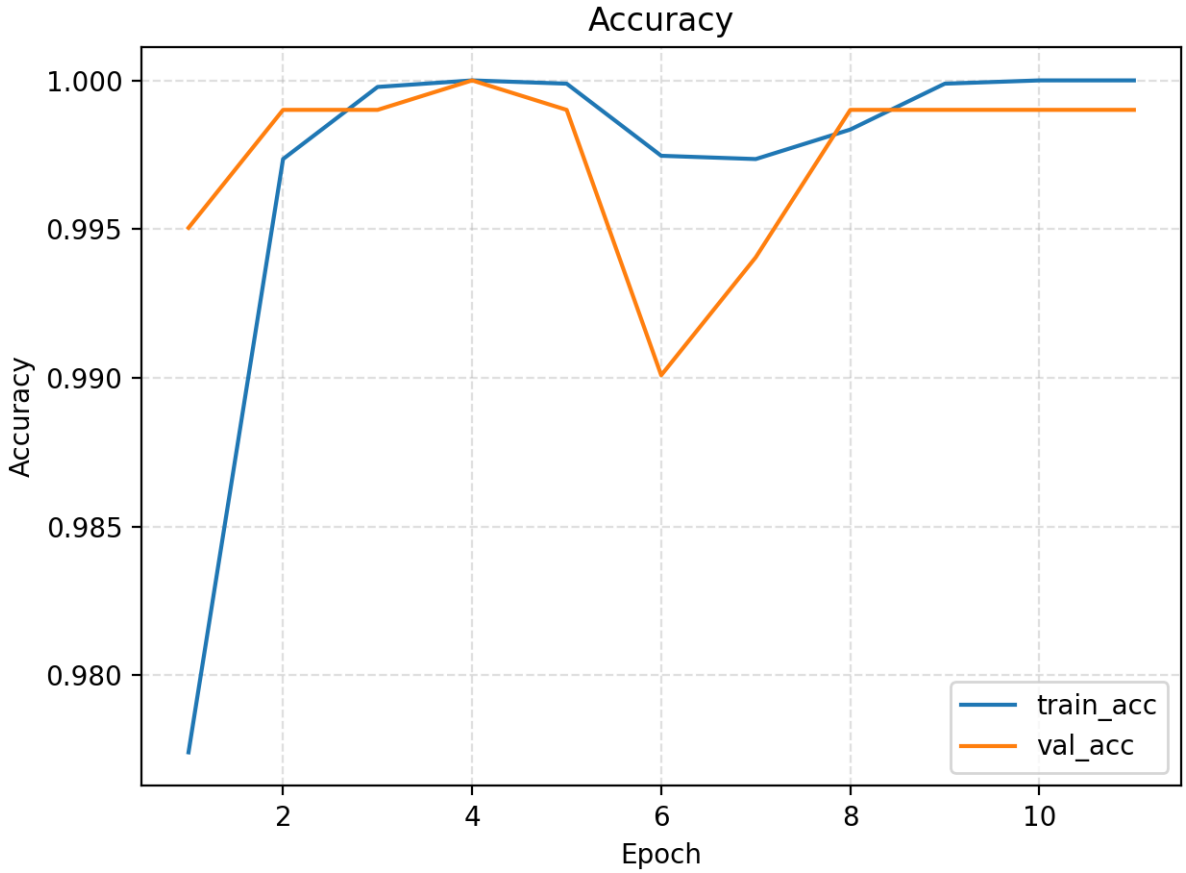


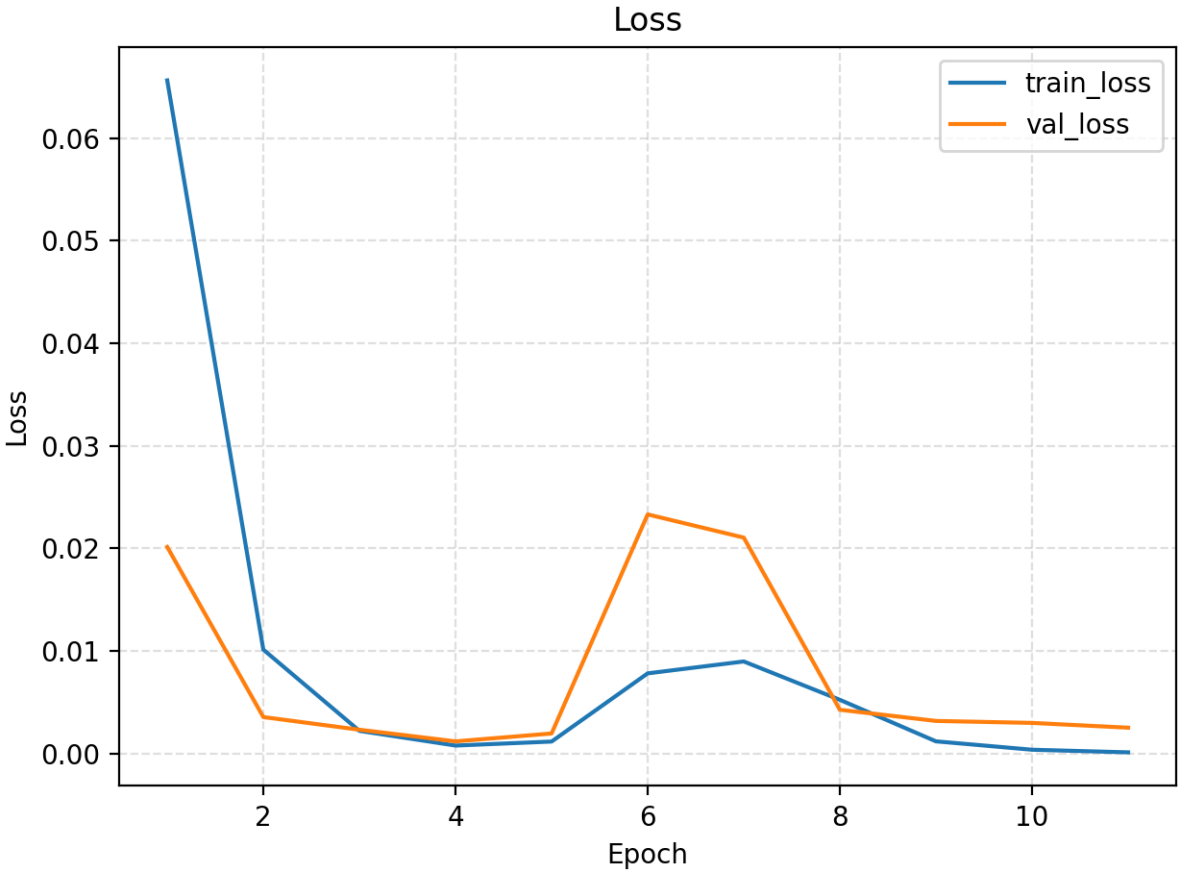
5.8. Model 7: Densenet121.pt

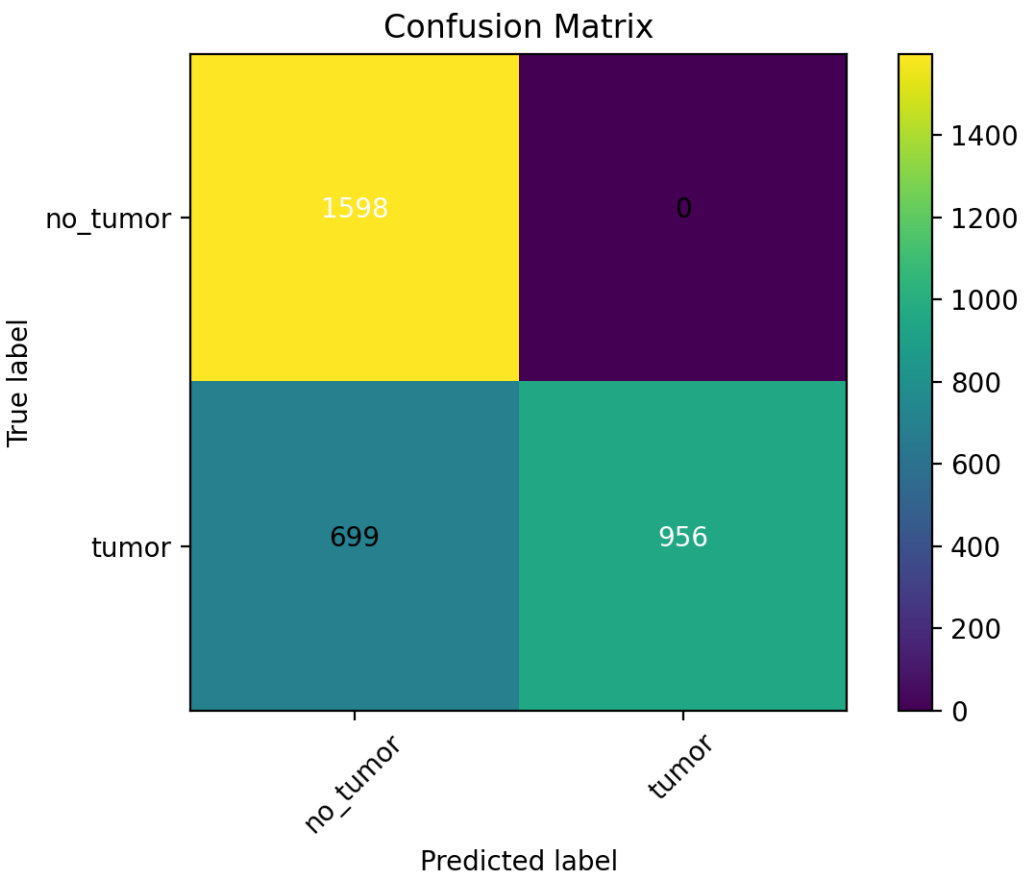
Sonuç Grafikleri

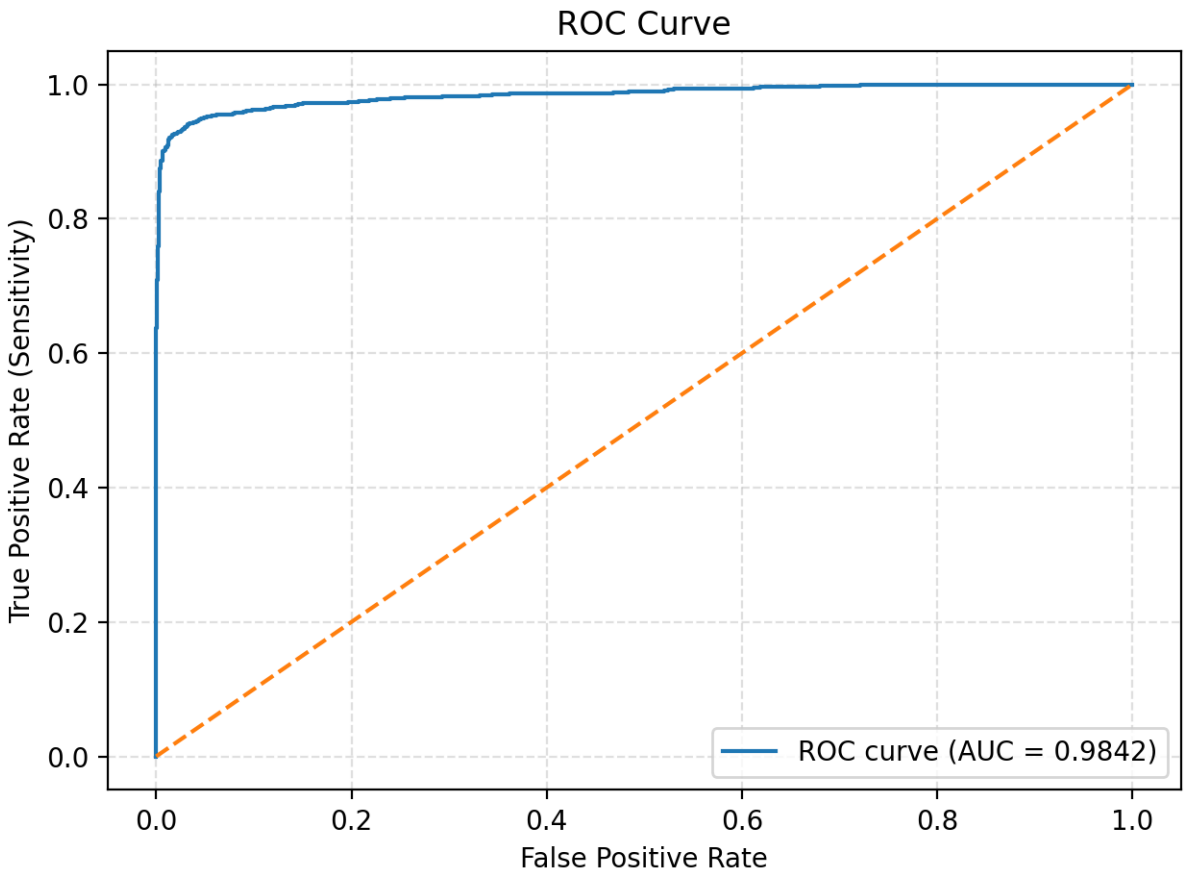
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.5776	0.7323	0.5733	0.7851


Accuracy
1.000
0.995
0.990
0.985
0.980
Accuracy
2
4
6
8
10
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Loss
2
4
6
8
10
Epoch


Confusion Matrix
1598
0
699
956
no_tumor
tumor
True label
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200
0

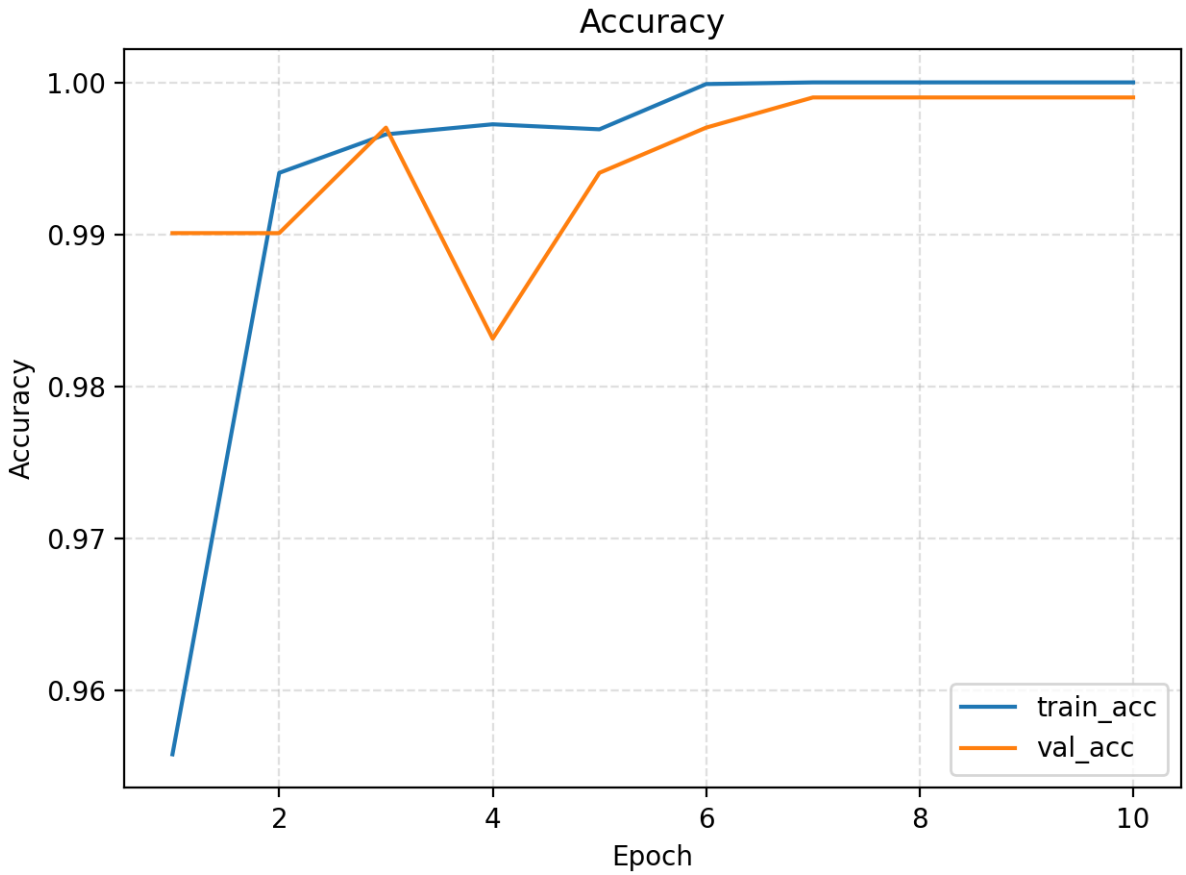

ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9842)

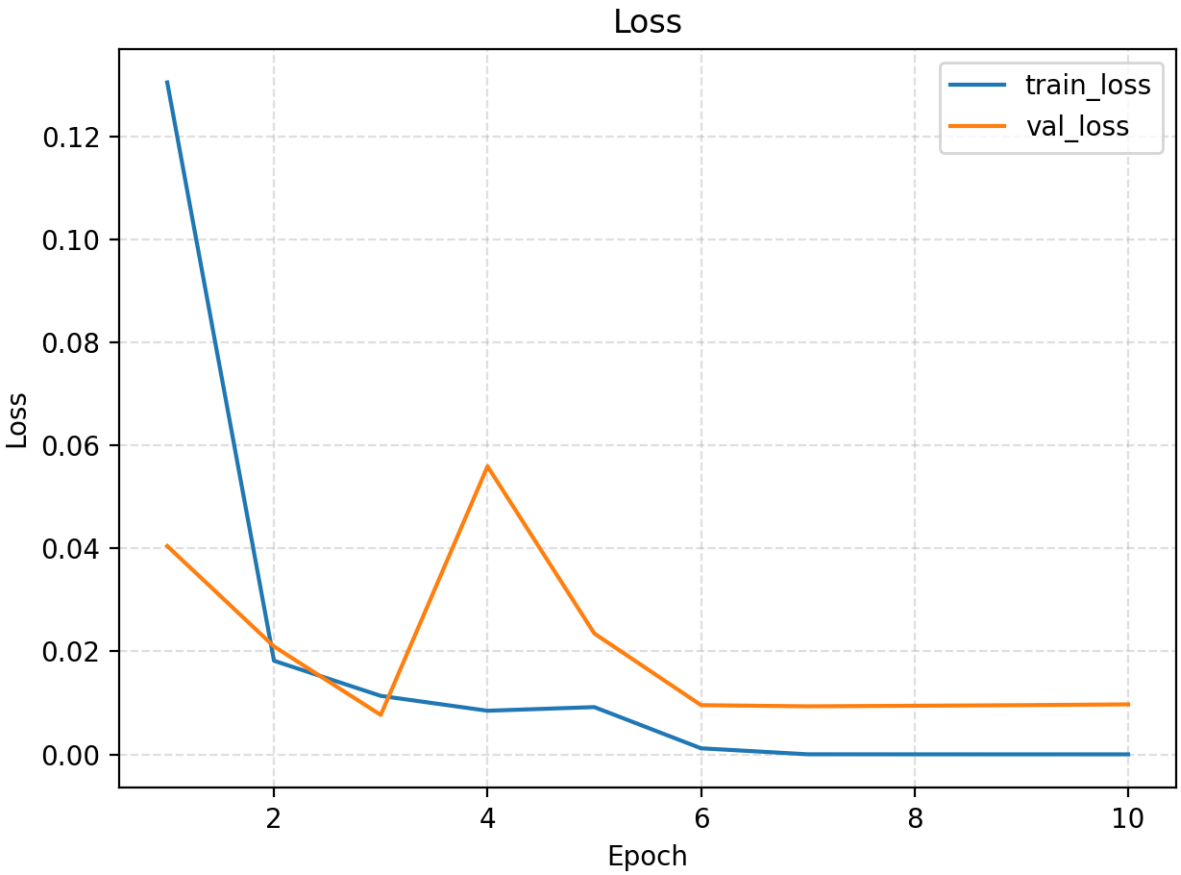
5.9. Model 8: convnext_tiny.pt

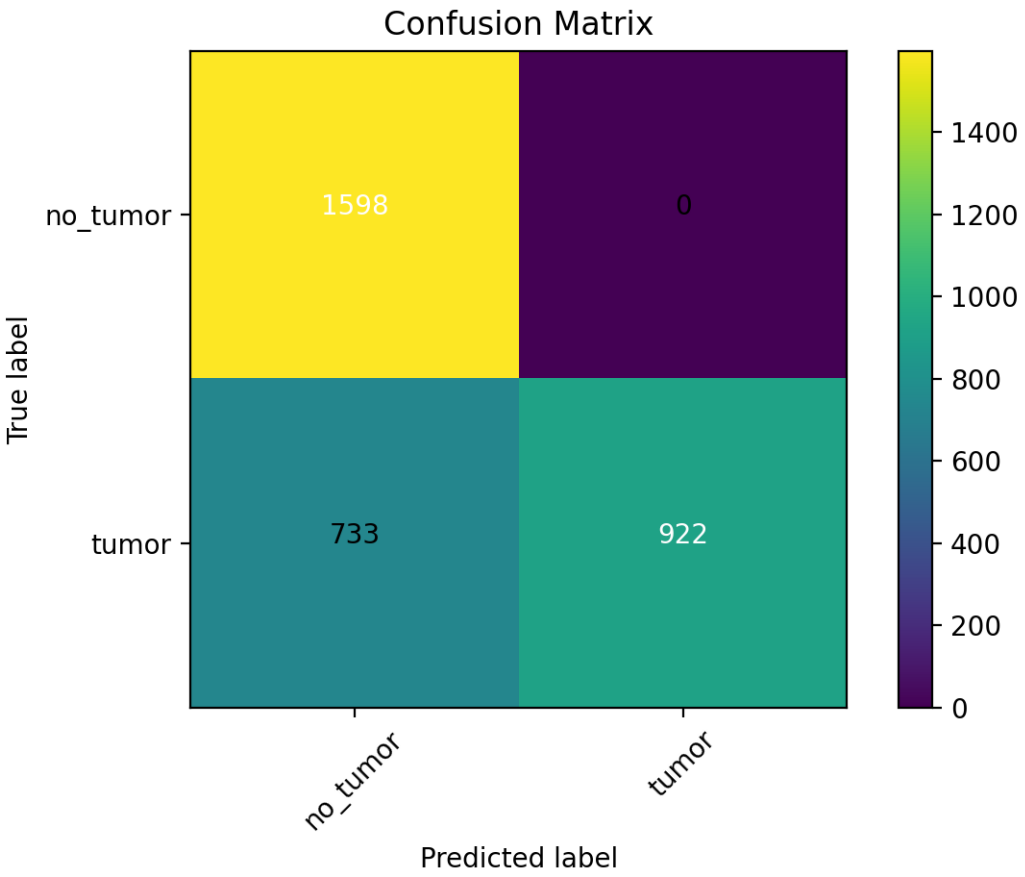
Sonuç Grafikleri

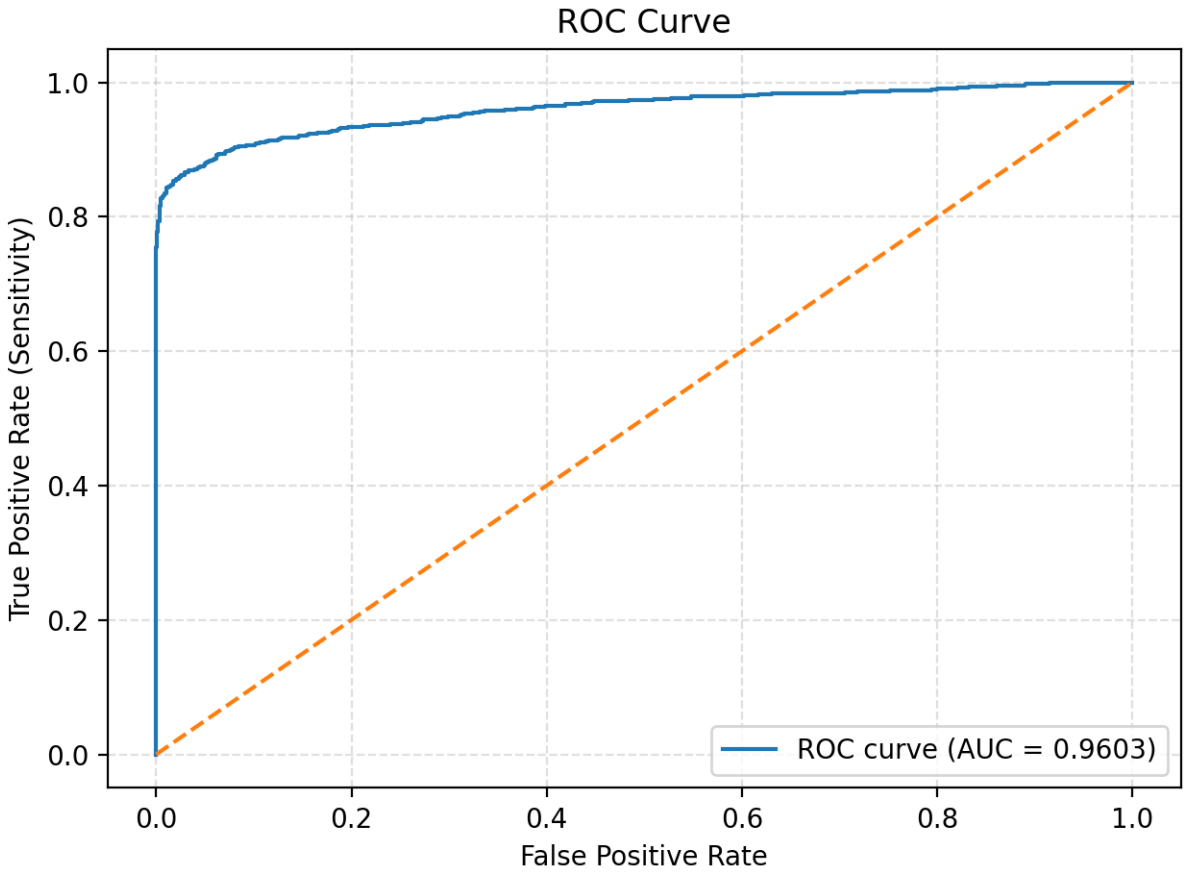
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.5571	0.7156	0.5527	0.7747


Accuracy
1.00
0.99
0.98
0.97
0.96
Accuracy
2
4
6
8
10
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.12
0.10
0.08
0.06
0.04
0.02
0.00
Loss
2
4
6
8
10
Epoch


Confusion Matrix
no_tumor
tumor
True label
1598
0
733
922
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200
0

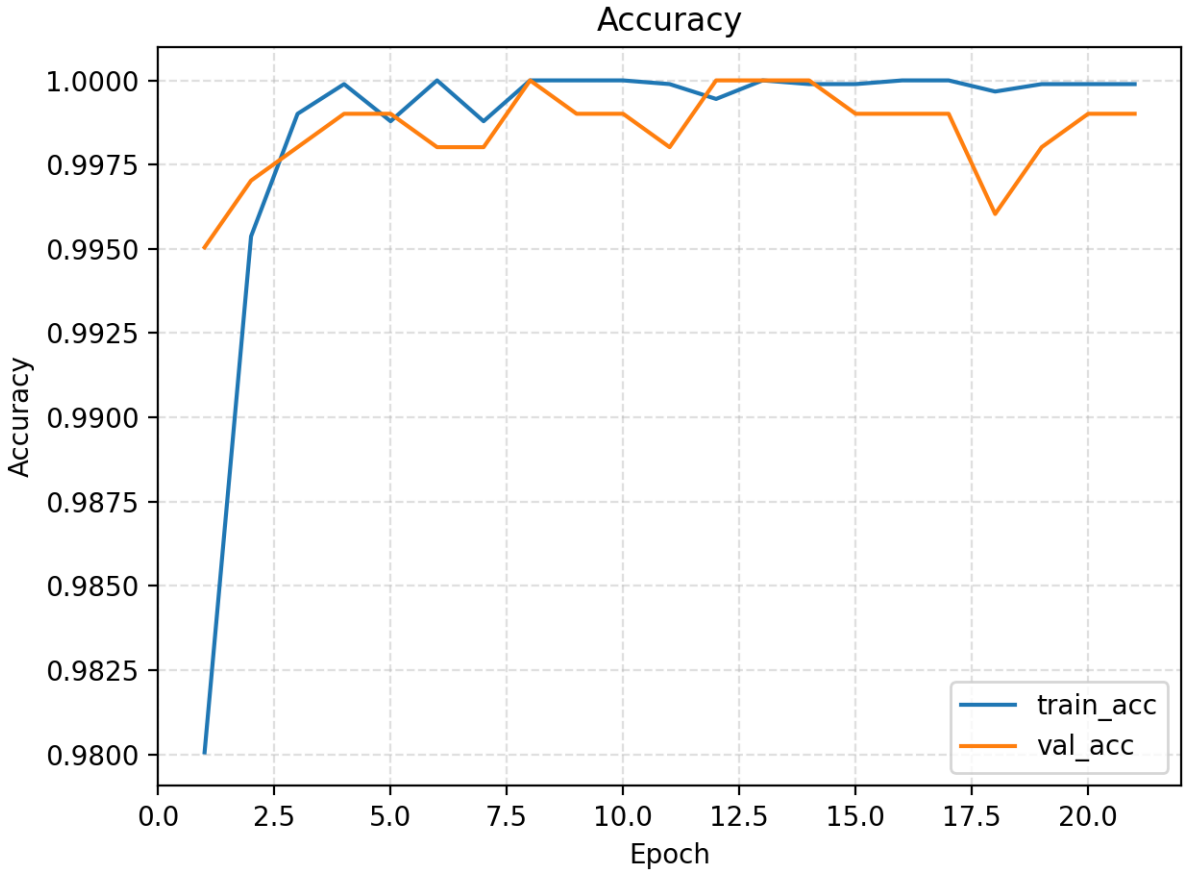


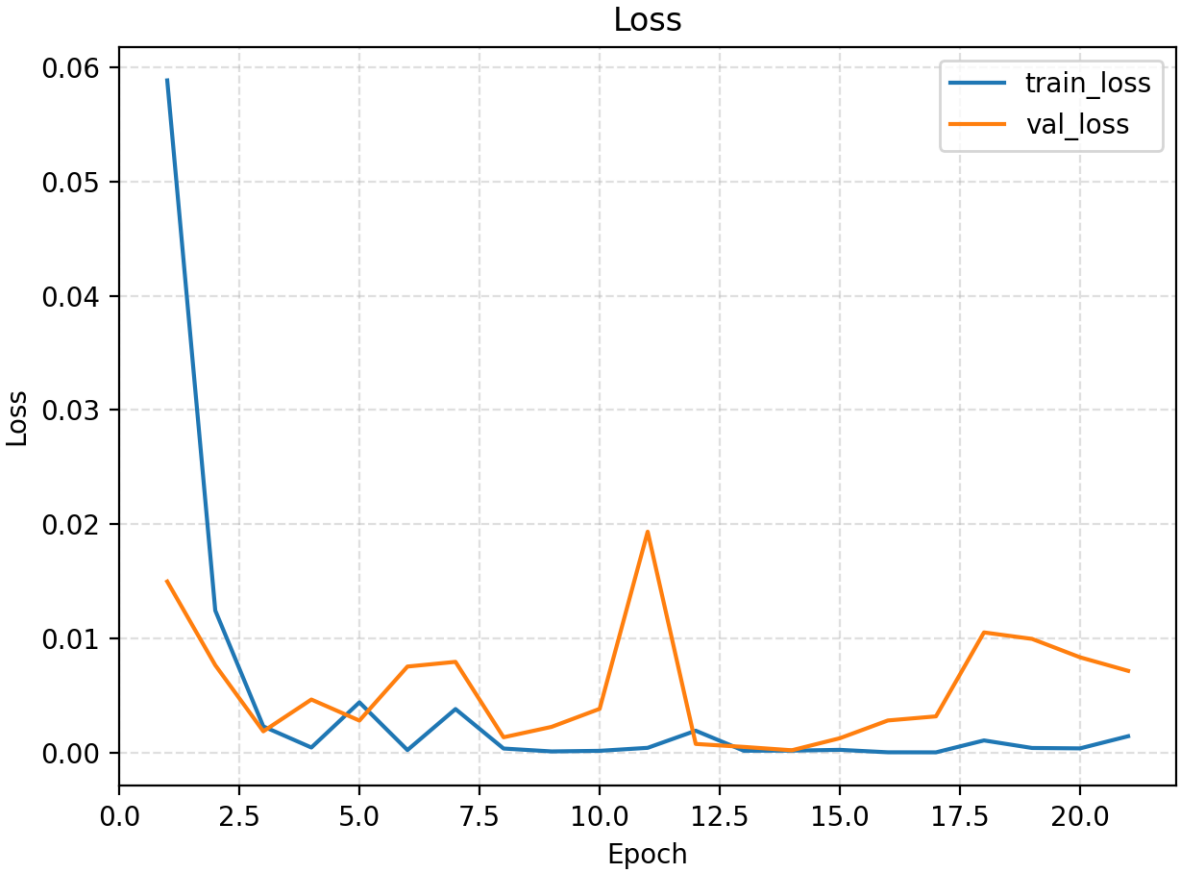
5.10. Model 9: hybrid_swinT_effb0.pt

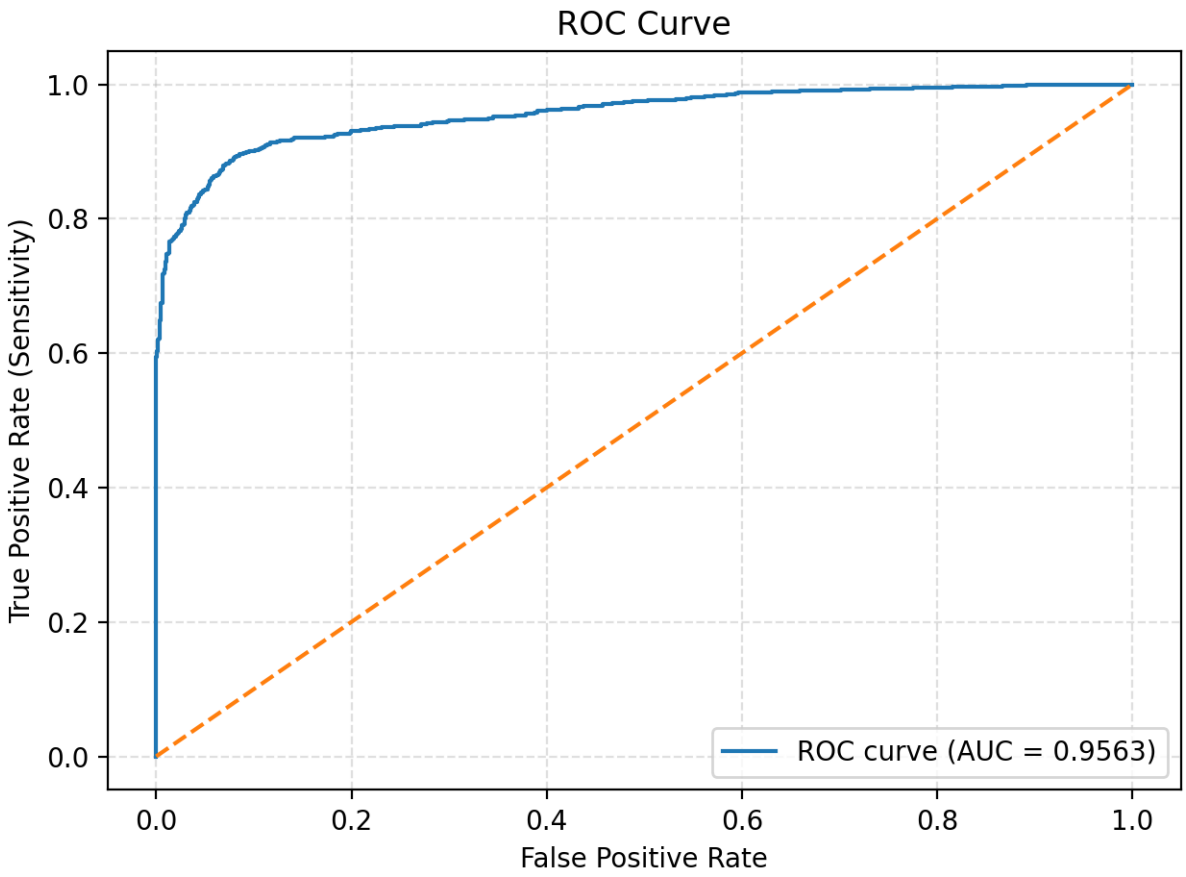
Sonuç Grafikleri

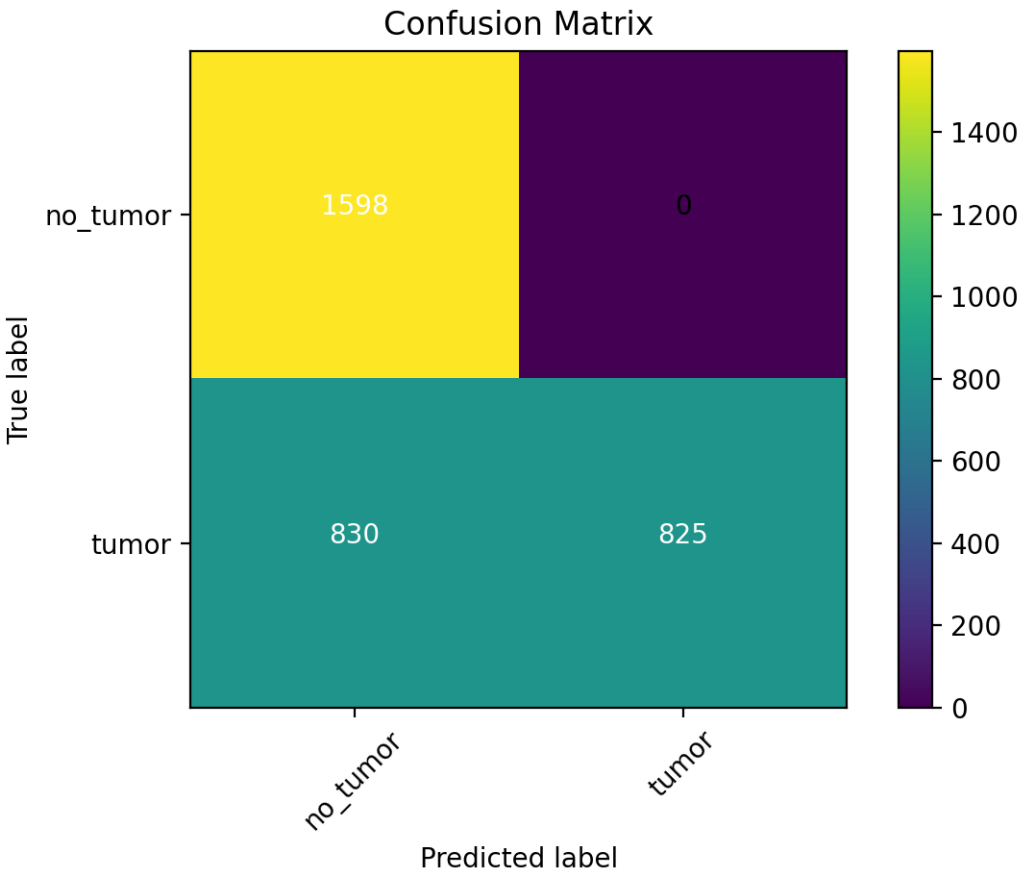
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.4985	0.6653	0.4941	0.7449


Accuracy
1.0000
0.9975
0.9950
0.9925
0.9900
0.9875
0.9850
0.9825
0.9800
Accuracy
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Loss
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9563)

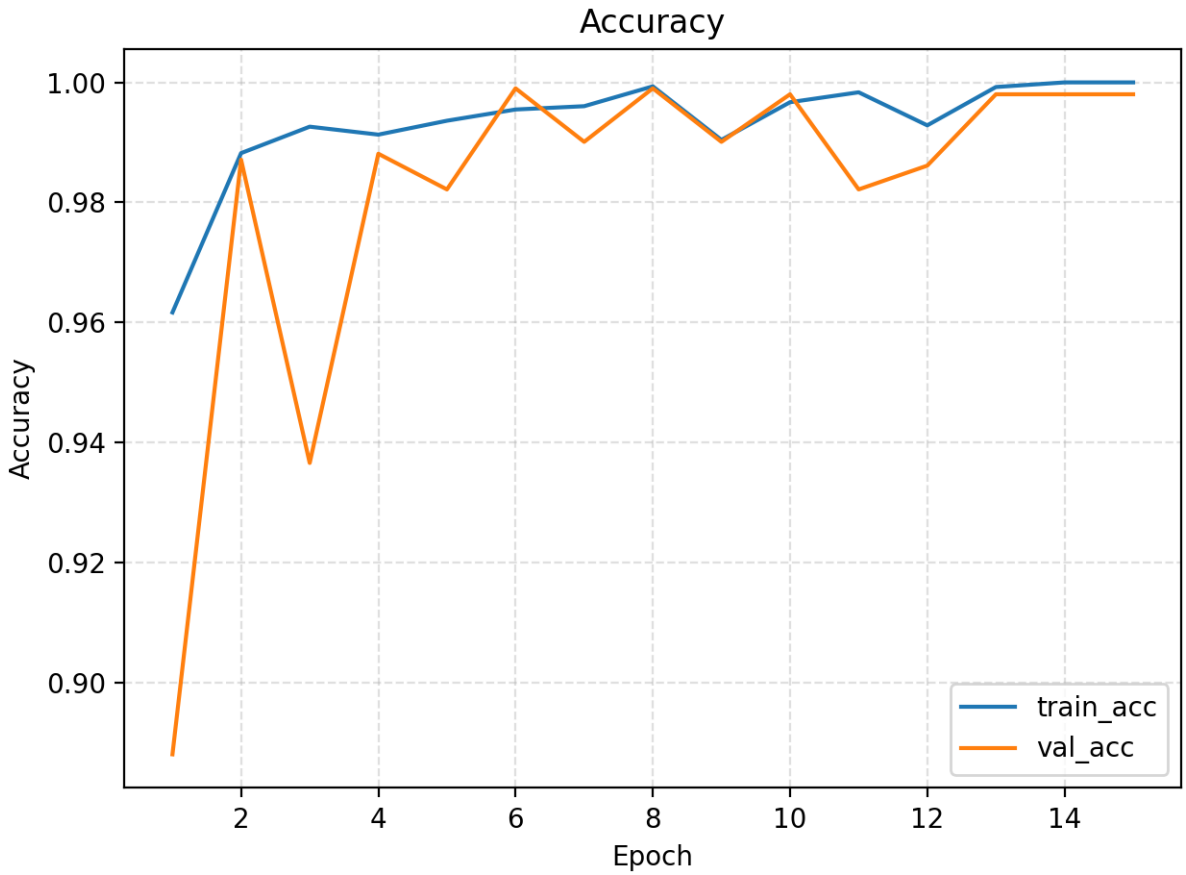

Confusion Matrix
no_tumor
tumor
True label
1598
0
830
825
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200
0

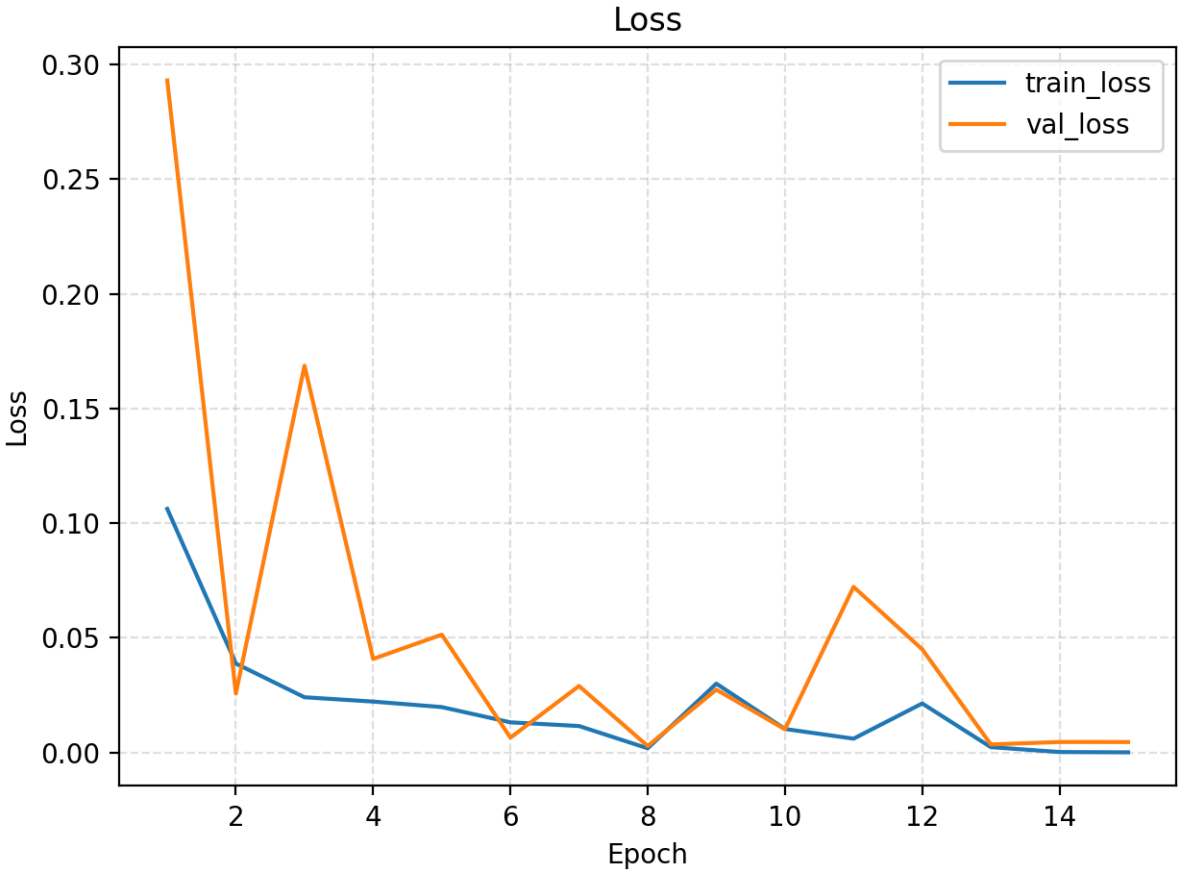
5.11. Model 10: resnet50.pt

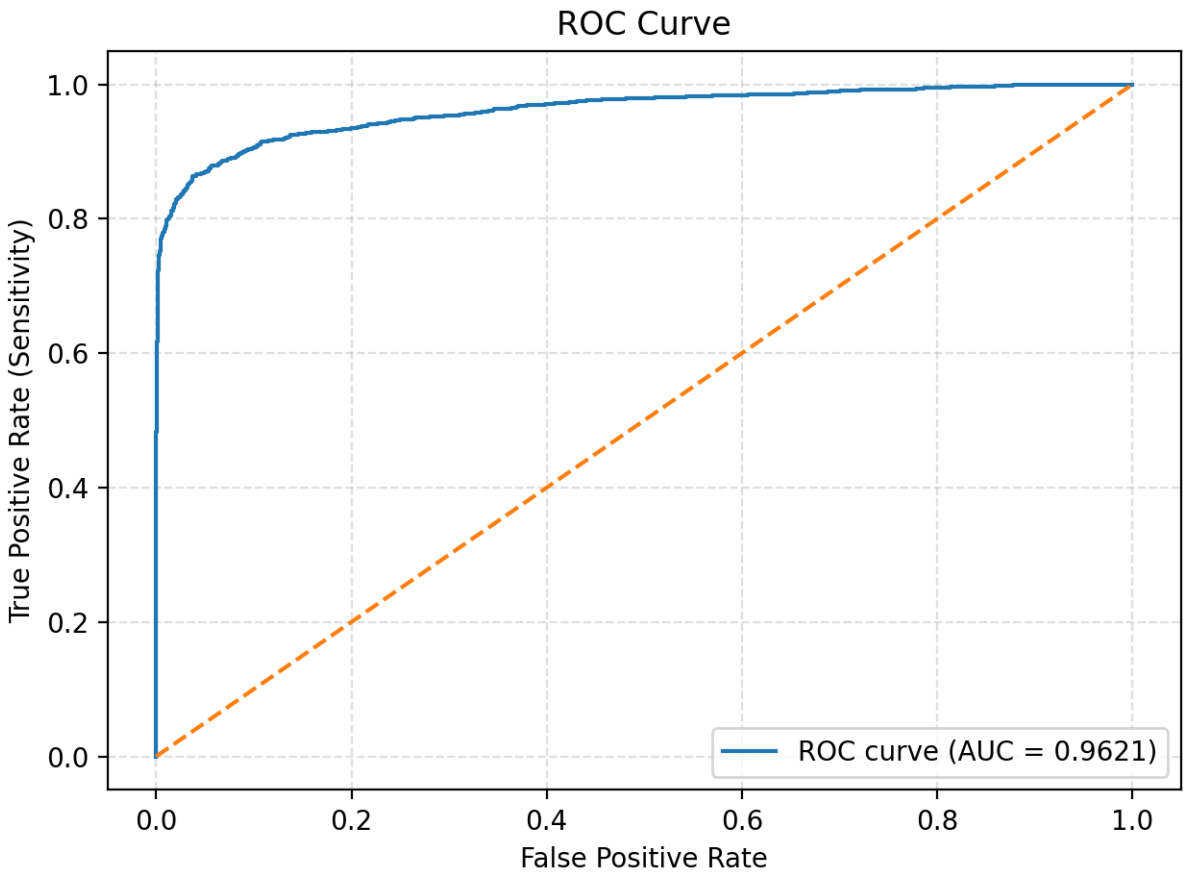
Sonuç Grafikleri

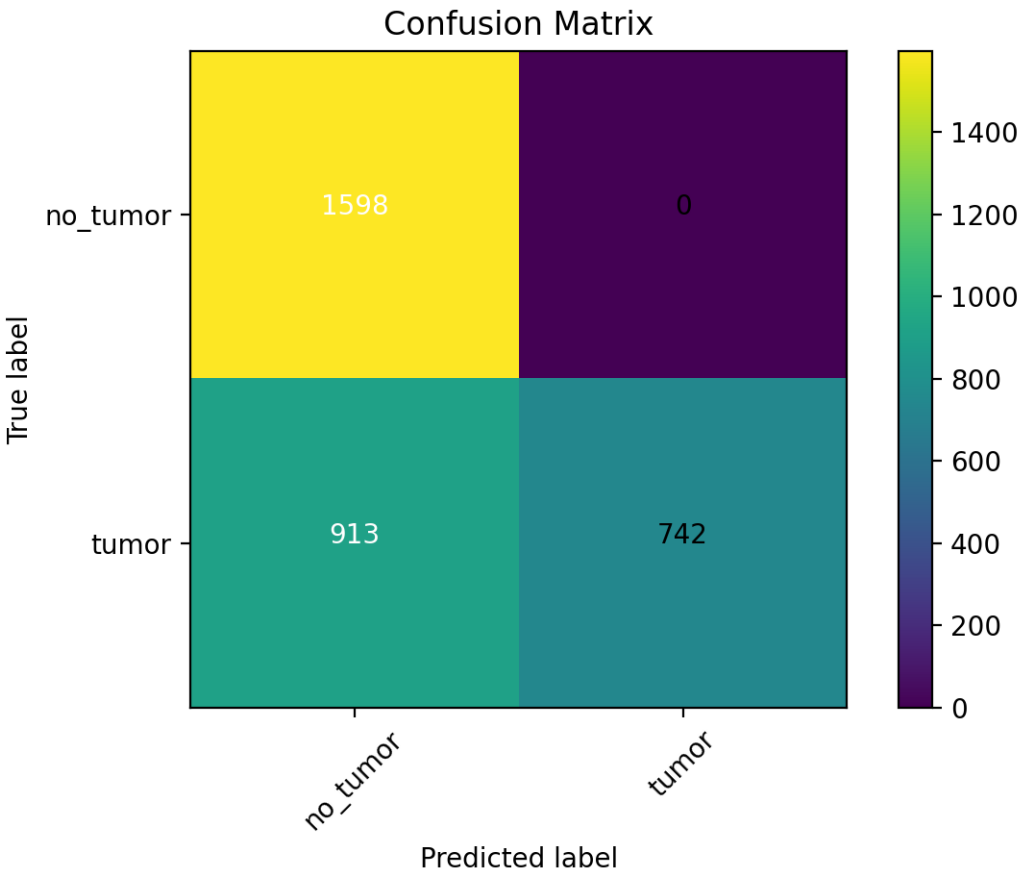
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.4483	0.6191	0.4440	0.7193


Accuracy
1.00
0.98
0.96
0.94
0.92
0.90
Accuracy
2
4
6
8
10
12
14
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.30
0.25
0.20
0.15
0.10
0.05
0.00
Loss
2
4
6
8
10
12
14
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9621)

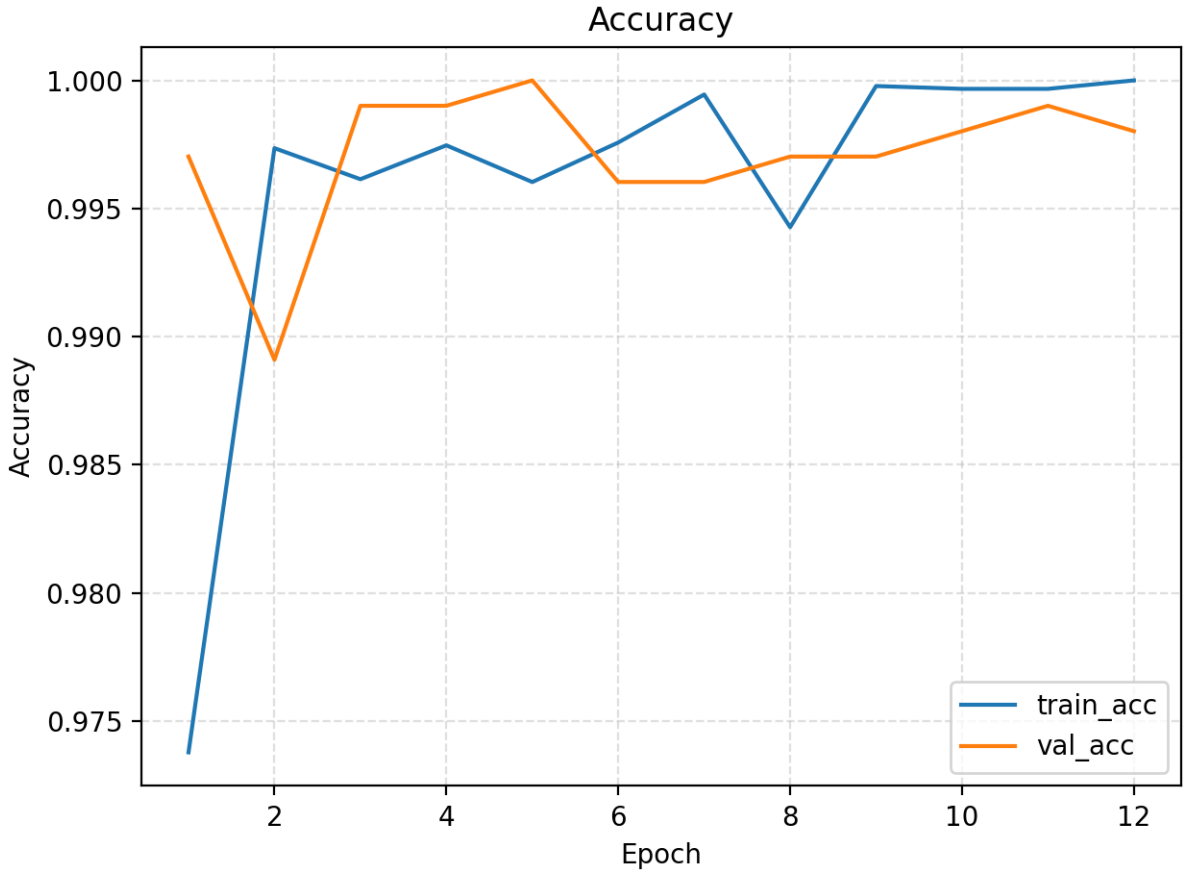

Confusion Matrix
no_tumor
tumor
True label
1598
0
913
742
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200
0

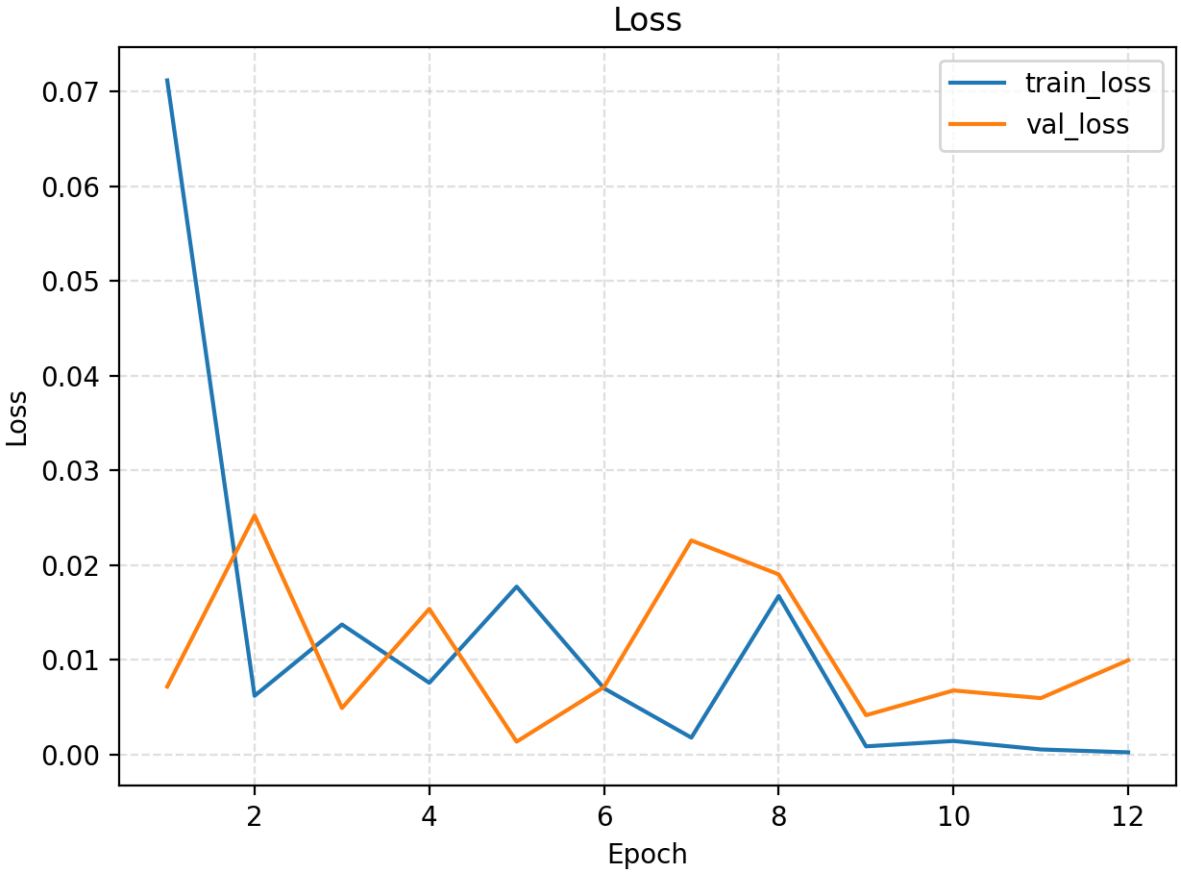
5.12. Model 10: inception_v3

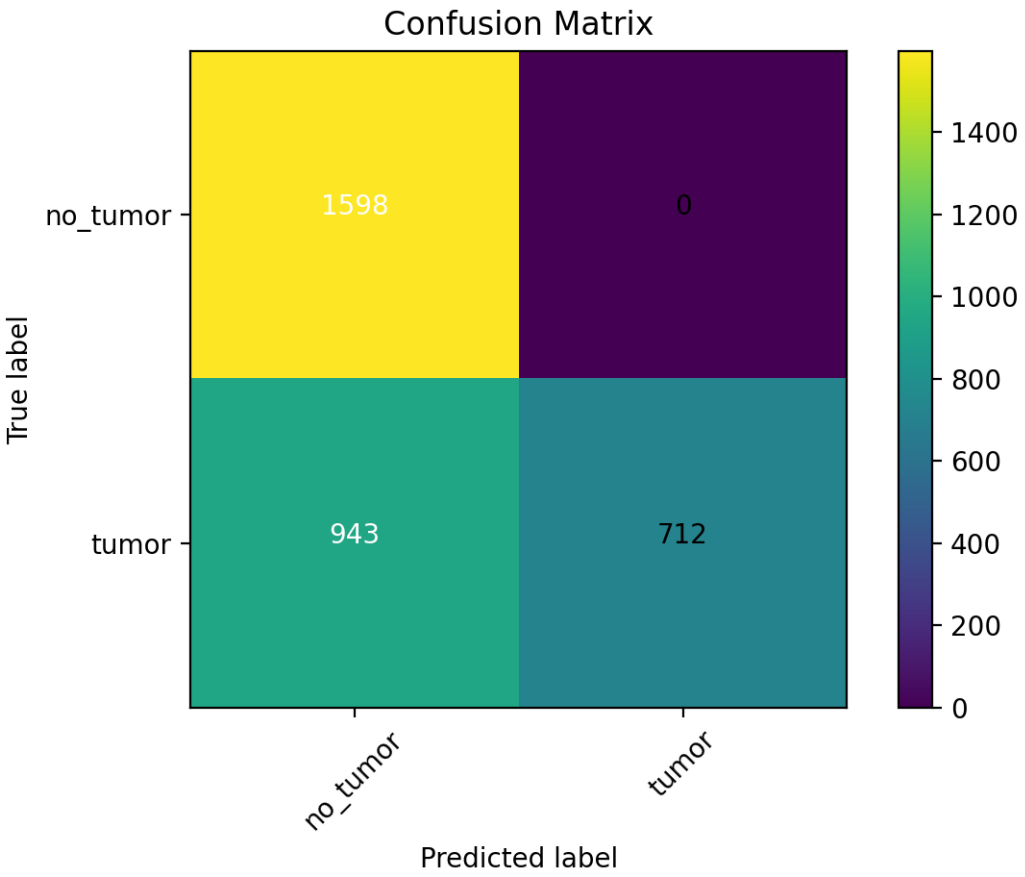
Sonuç Grafikleri

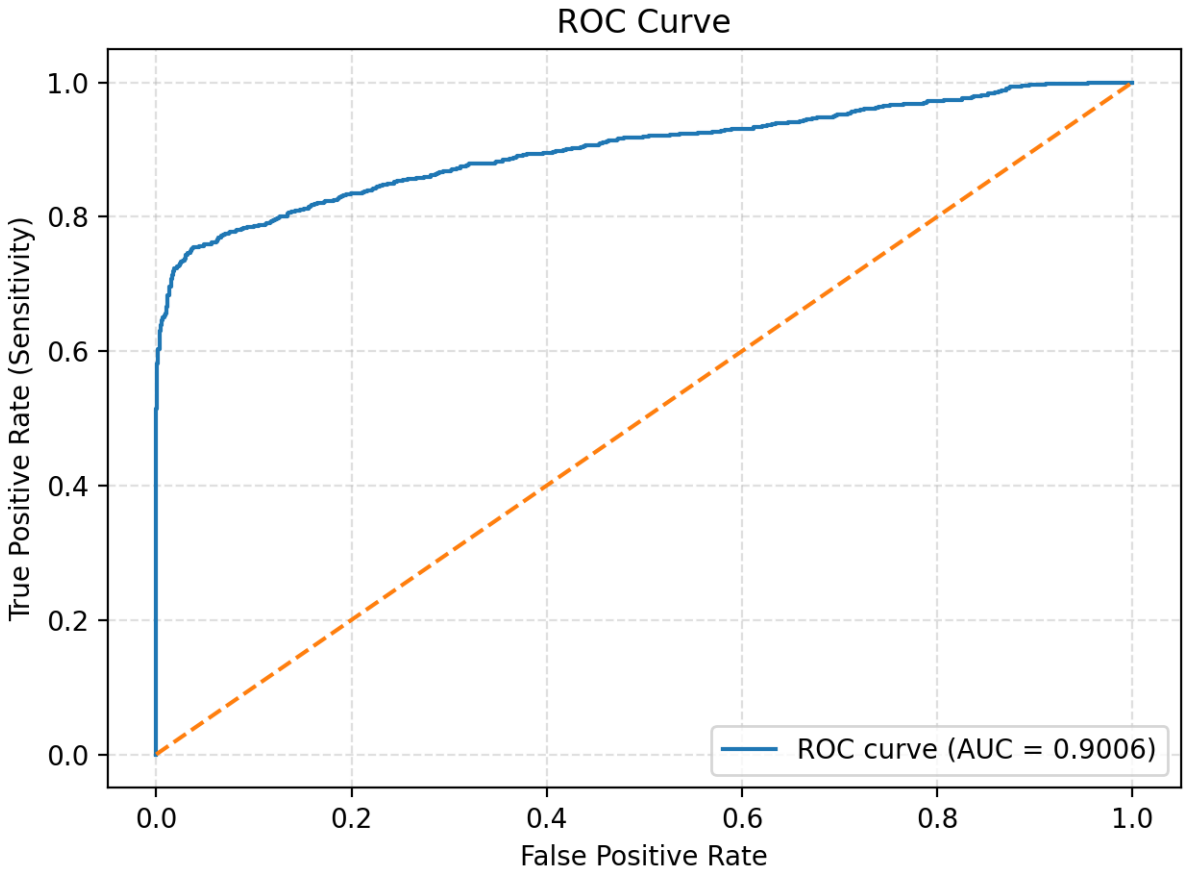
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.4302	0.6016	0.4259	0.7101


Accuracy
1.000
0.995
0.990
0.985
0.980
0.975
Accuracy
2
4
6
8
10
12
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
Loss
2
4
6
8
10
12
Epoch


Confusion Matrix
no_tumor
tumor
True label
1598
0
943
712
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200
0

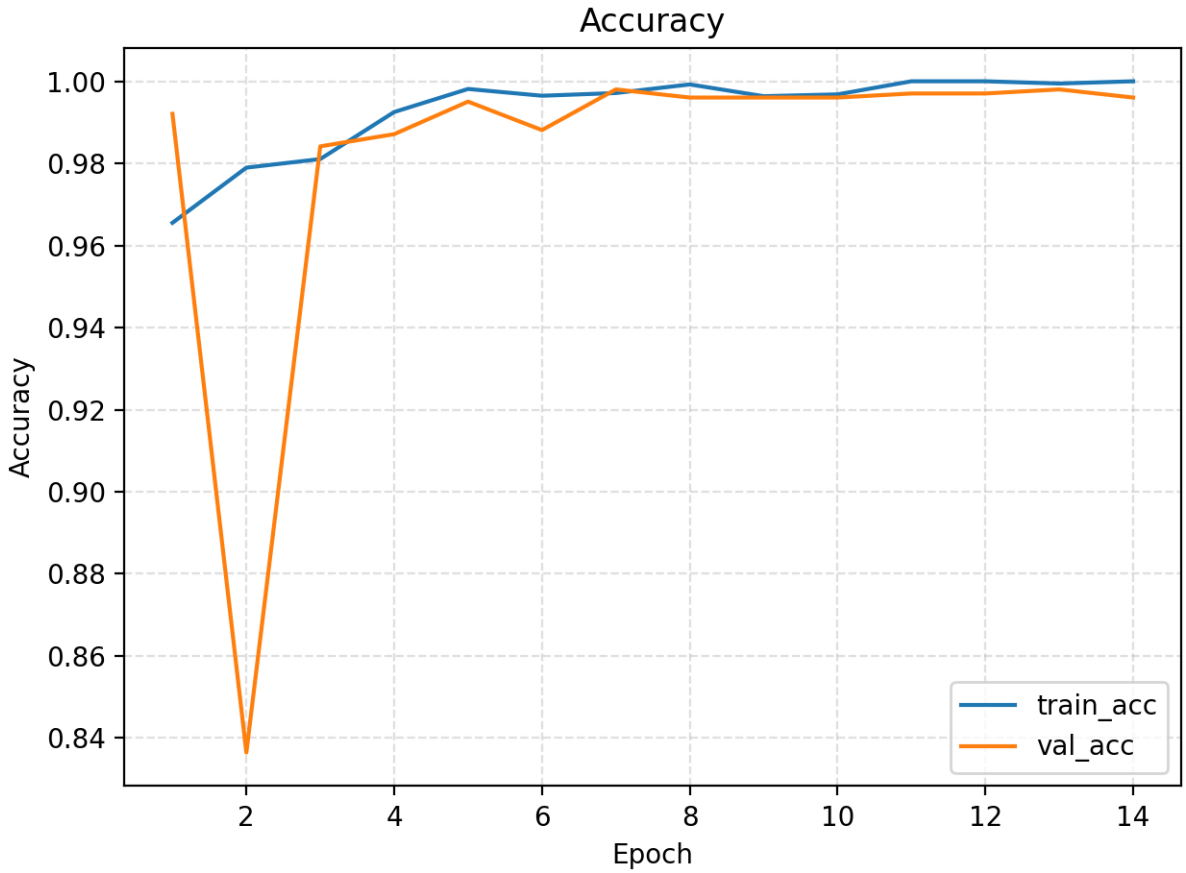


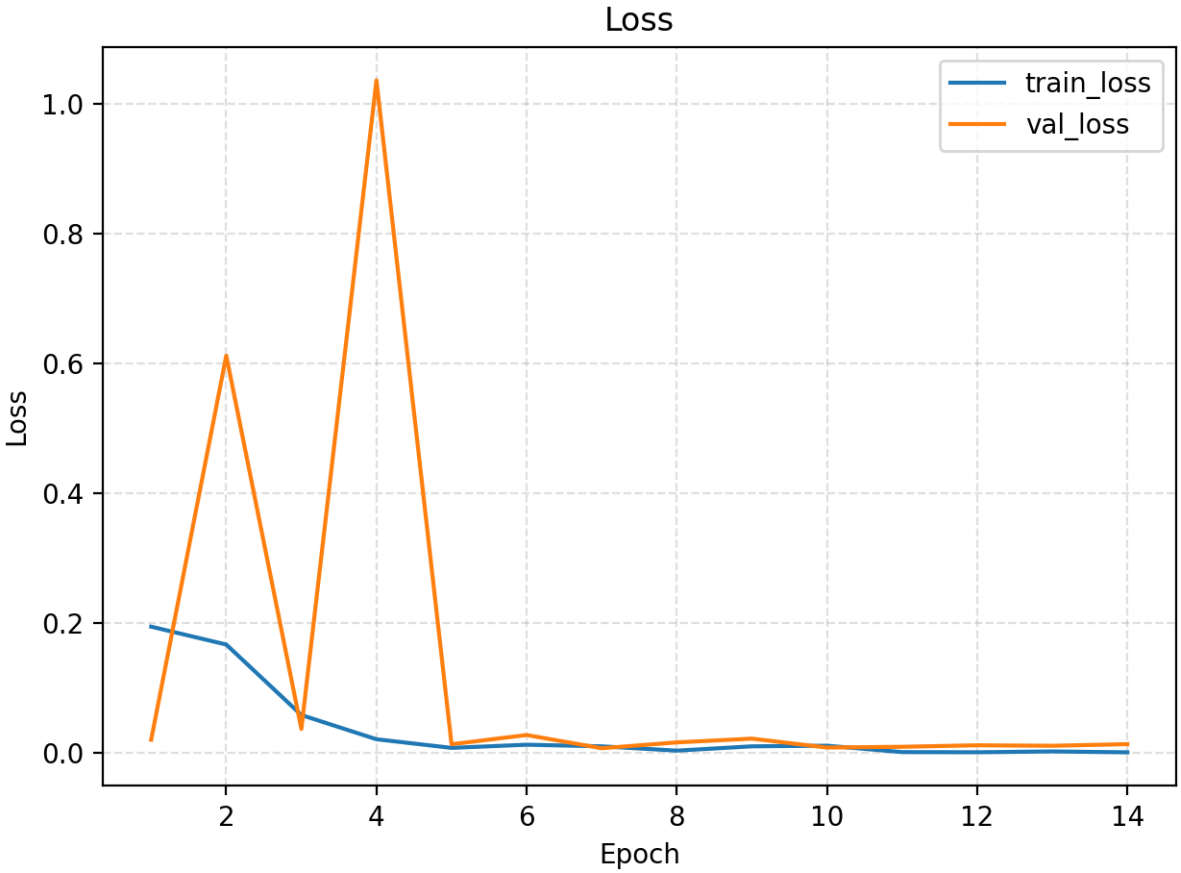
5.14. Model 10: efficientnet_b0.pt

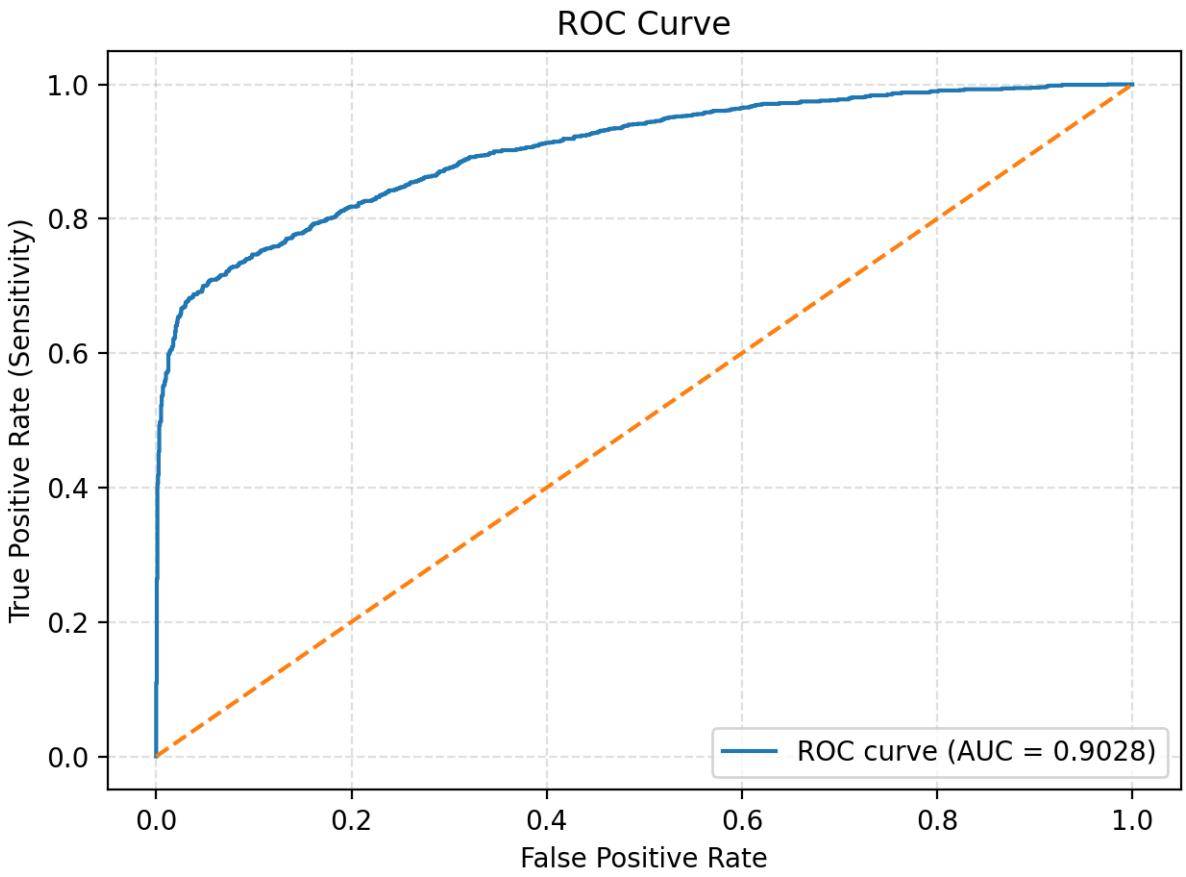
Sonuç Grafikleri

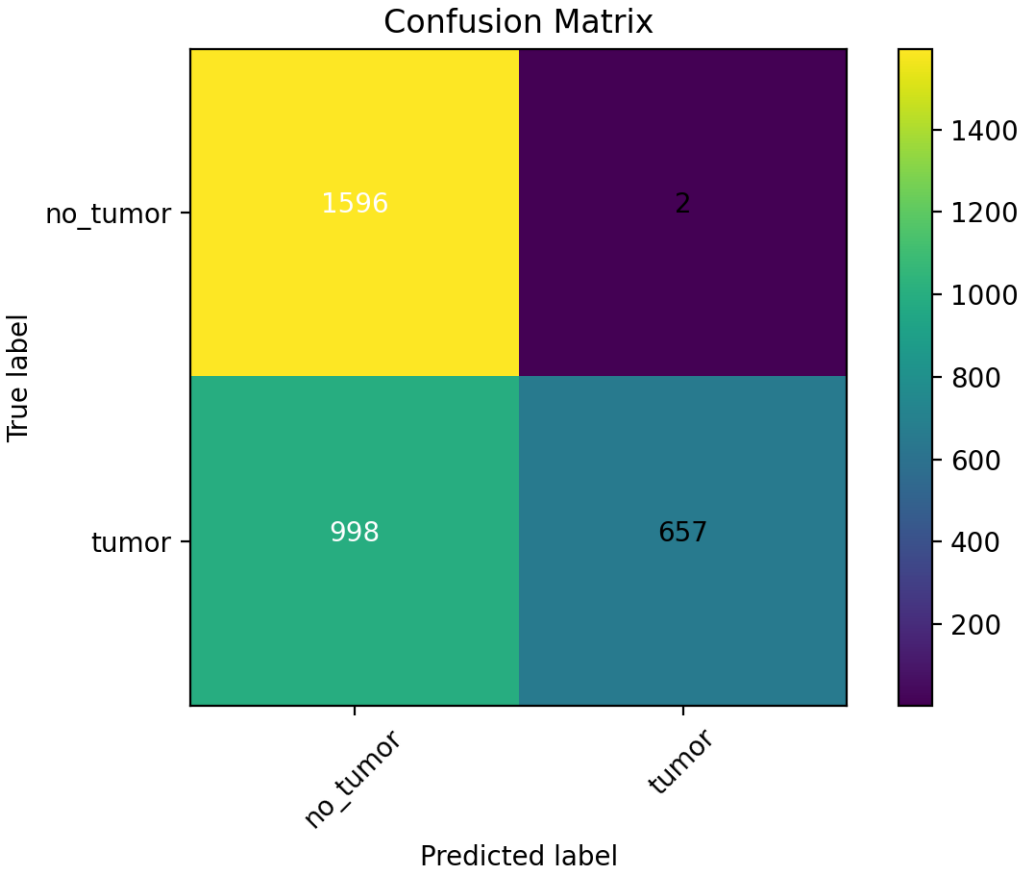
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
0.9970	0.3970	0.5678	0.3915	0.6926




Loss
train_loss
val_loss
1.0
0.8
0.6
0.4
0.2
0.0
Loss
2
4
6
8
10
12
14
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.9028)

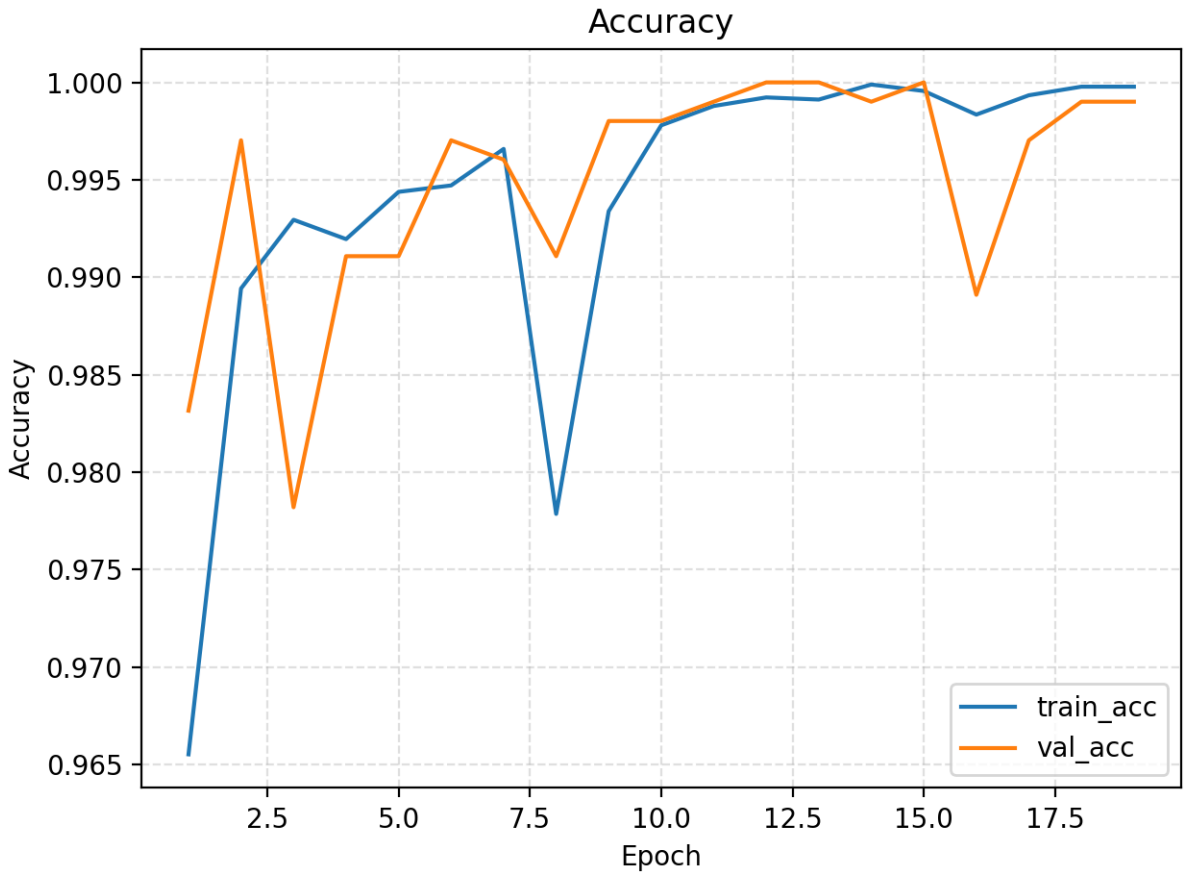

Confusion Matrix
1596
2
998
657
no_tumor
tumor
True label
no_tumor
tumor
Predicted label
1400
1200
1000
800
600
400
200

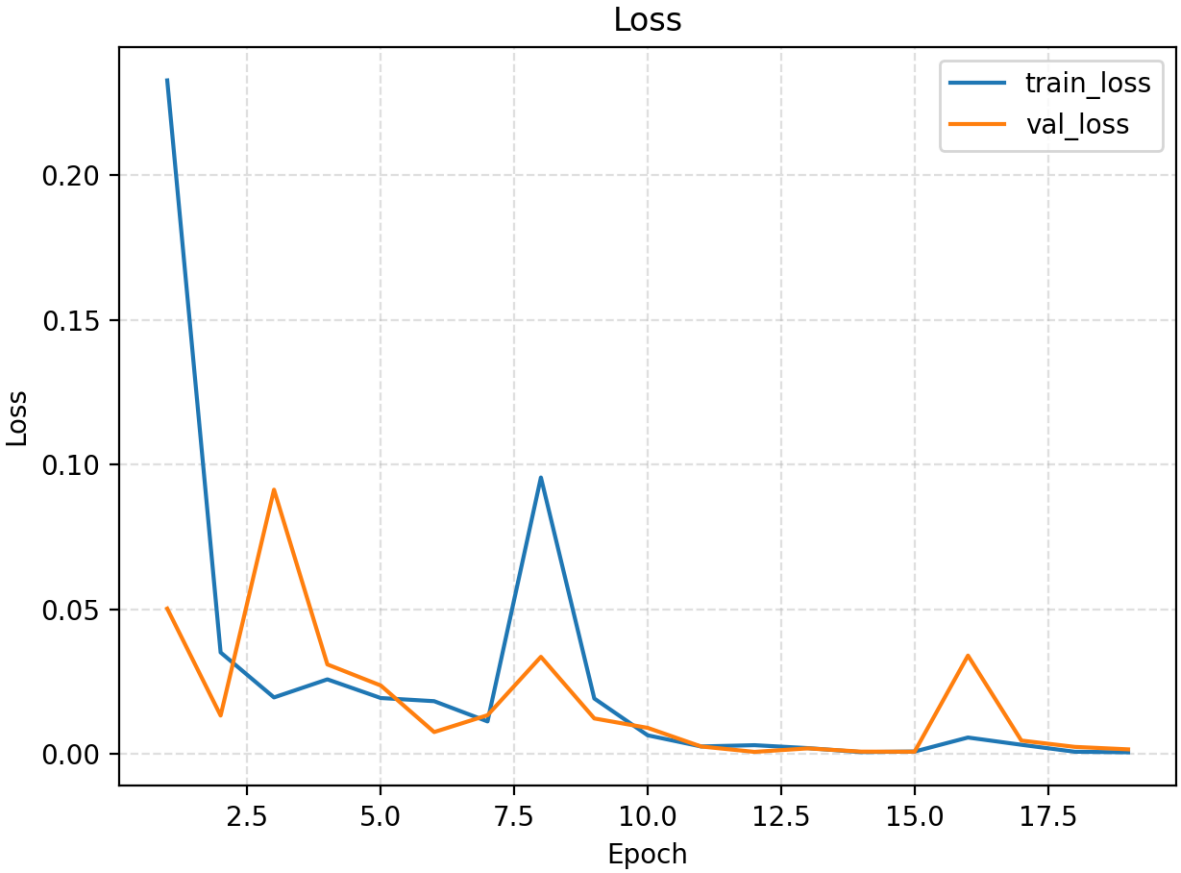
5.15. Model 10: mobilenetv2_100.pt

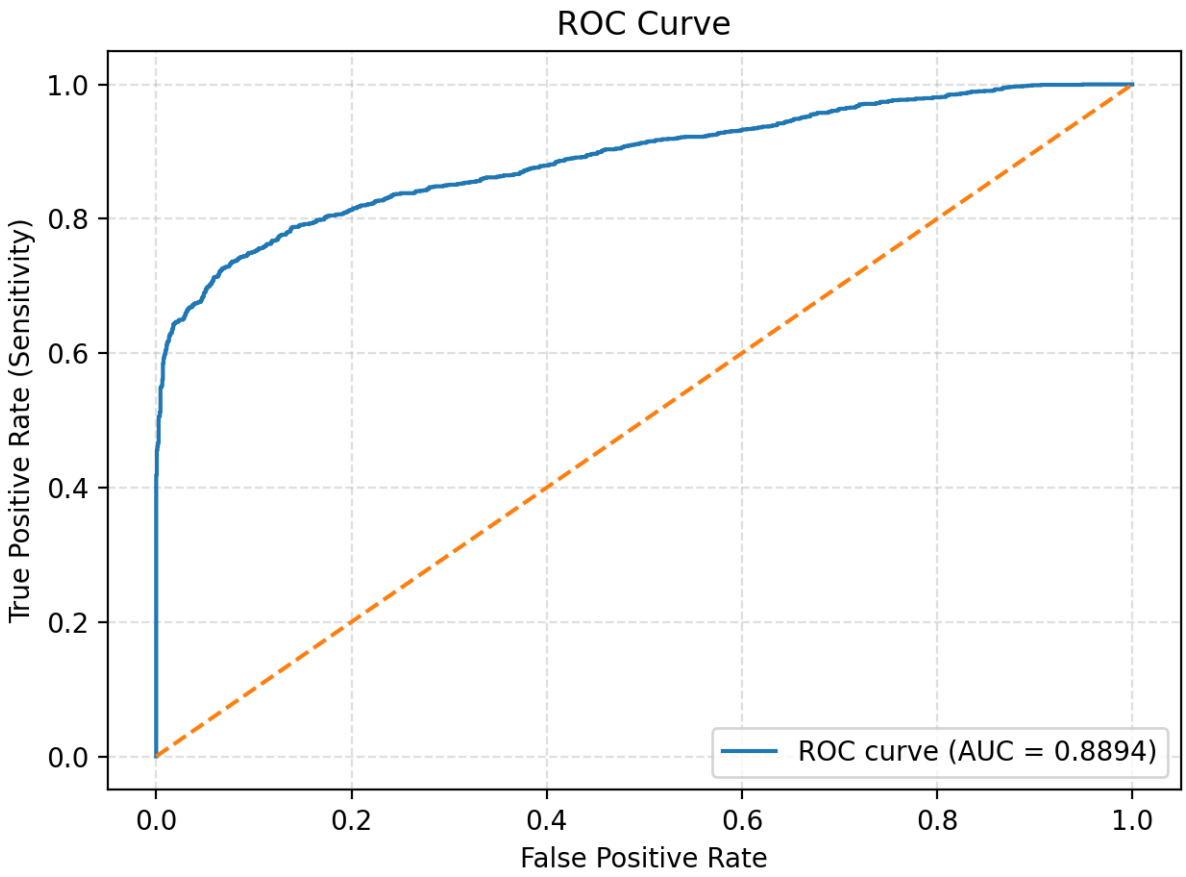
Sonuç Grafikleri

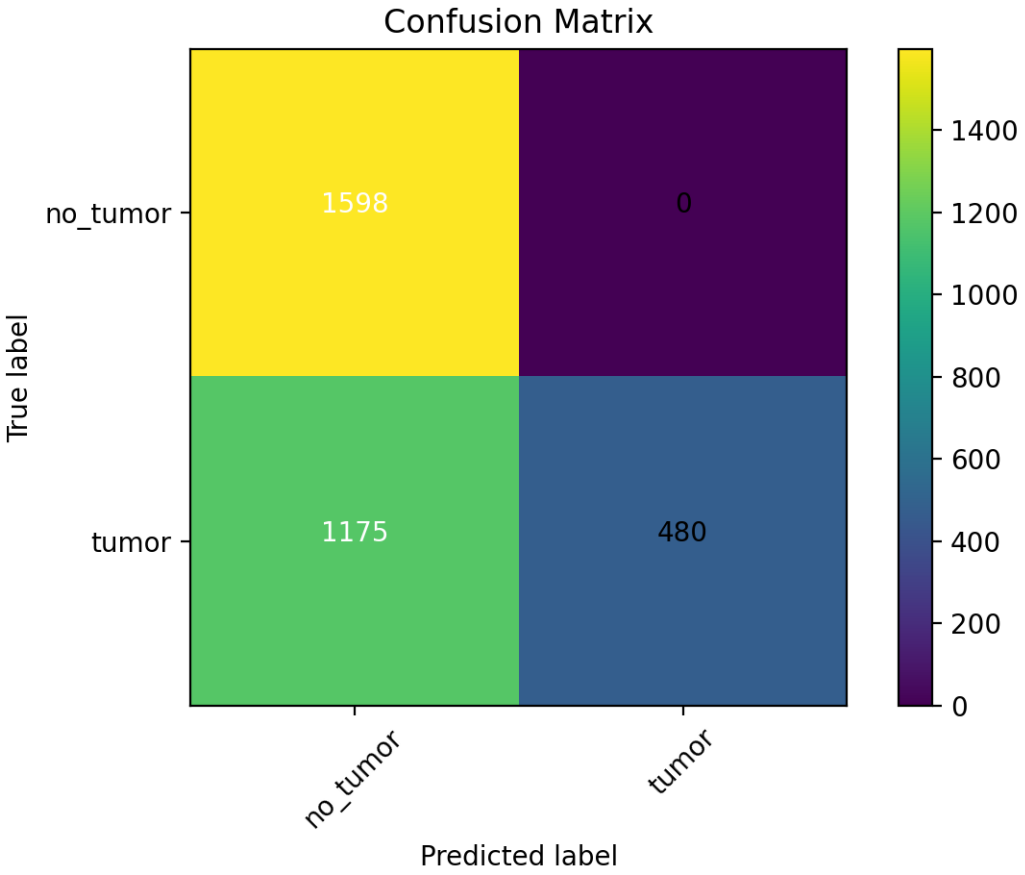
Değerlendirme Metrikleri

Precision	Sensitivity (Recal	F1-Score	Cohen's Kappa	Accuracy
1.0000	0.2900	0.4496	0.2864	0.6388


Accuracy
1.000
0.995
0.990
0.985
0.980
0.975
0.970
0.965
Accuracy
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch
train_acc
val_acc


Loss
train_loss
val_loss
0.20
0.15
0.10
0.05
0.00
Loss
2.5
5.0
7.5
10.0
12.5
15.0
17.5
Epoch


ROC Curve
True Positive Rate (Sensitivity)
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (AUC = 0.8894)



6. Kaynaklar

Mendeley eğitim veri seti: https://data.mendeley.com/datasets/c9rt8d6zrf/1

Kaggle test veri seti 1: https://www.kaggle.com/datasets/navoneel/brain-mri-images-for-brain-tumor-detection

Kaggle test veri seti 2: https://www.kaggle.com/datasets/ahmedhamada0/brain-tumor-detection