Üretici Modellerin Genel Tanımı ve Önemi

Üretici model nedir?

- Elinde bir veri kümesi var (örneğin yüz fotoğrafları, kediler, manzaralar, ses kayıtları, metinler).
- Bu veri kümesinden "veri nasıl görünür / ses çıkarır / yazılır?" bilgisini öğrenen ve bu bilgiye dayanarak yeni, benzer fakat birebir aynı olmayan örnekler üreten modellere üretici model denir.
- Yani amaç:"Bana bu veri tipinden yenilerini üret."

Ayırt edici modellerden farkı

- Ayırt edici model: "Bu kedi mi, köpek mi?" diye karar verir.
- Üretici model: "Bana yeni bir kedi resmi üret." diyebilirsin.

Üretici Modellerin Genel Tanımı ve Önemi

Neden önemliler

- Gerçek veriye çok benzeyen sahte veri üretilebilir.

- Veri üretimi sayesinde;
  - Küçük veri setlerini zenginleştirebilirsin (data augmentation).
  - Gerçek veriye ulaşmanın zor ya da pahalı olduğu senaryolarda sentetik veri kullanabilirsin.
  - Yaratıcı alanlarda (sanat, tasarım, müzik) tamamen yeni içerikler oluşturulabilir.

- Günümüzdeki birçok "gösterişli" yapay zeka uygulamasının arkasında üretici modeller vardır(metinden resim, metinden müzik, gerçekçi yüz üretimi, derin sahte – deepfake, vb.).

Yapay Zekâda Üretici Modellerin Yeri ve Kullanım Alanları

- **Görsel alan (bilgisayarlı görü)**
  - Gerçekçi insan yüzleri (örneğin hiç var olmayan insanlar).
  - Oda, manzara, ürün, moda tasarımı görselleri.
  - Metinden görsele: "Uzayda gitar çalan kedi" yazıp görüntü oluşturma.
  - Veri artırma:
    - Örneğin tümör tespiti için tıbbi görüntü sayın azsa, ek sentetik görüntü üretilebilir.
- **Metin üretimi**
  - Özet çıkarma, soru-cevap, hikâye yazma gibi alanlarda
  - Dil modelleri de geniş anlamda üretici modellerdir.
  - Örneğin: ChatGPT tarzı sistemler, bir tür üretici model ailesine dahildir.
- **Ses ve konuşma**
  - Metinden konuşma (Text-to-Speech) sistemleri.
  - Bir kişinin sesine benzeyen sentetik ses üretimi.
  - Müzik üretimi, ortam sesleri (yağmur, kalabalık, doğa sesleri) üretimi.

Yapay Zekâda Üretici Modellerin Yeri ve Kullanım Alanları

- **Oyun, film ve medya**
  - Oyun içi ortam ve karakter tasarımlarının otomatik üretilmesi.
  - Film sahnelerinde gerçekçi arka planlar, dijital dublörler.
  - Deepfake videolar (hem fırsat hem risk).
- **Sağlık ve bilimsel simülasyon**
  - Tıbbi görüntüler (MR, röntgen, CT) için sentetik veri.
  - Molekül ve ilaç tasarımında olası moleküler yapıların üretilmesi.
  - Fiziksel süreçlerin (akış, hareket) simülasyonu için sahte ama gerçekçi veri.
- **Gizlilik ve anonimleştirme**
  - Gerçek kişilere ait veriler yerine, onlara çok benzeyen sentetik veri kullanarak hem analiz yapmak hem de gizliliği korumak mümkün.

Neden Özellikle GAN ve Difüzyon Modelleri?

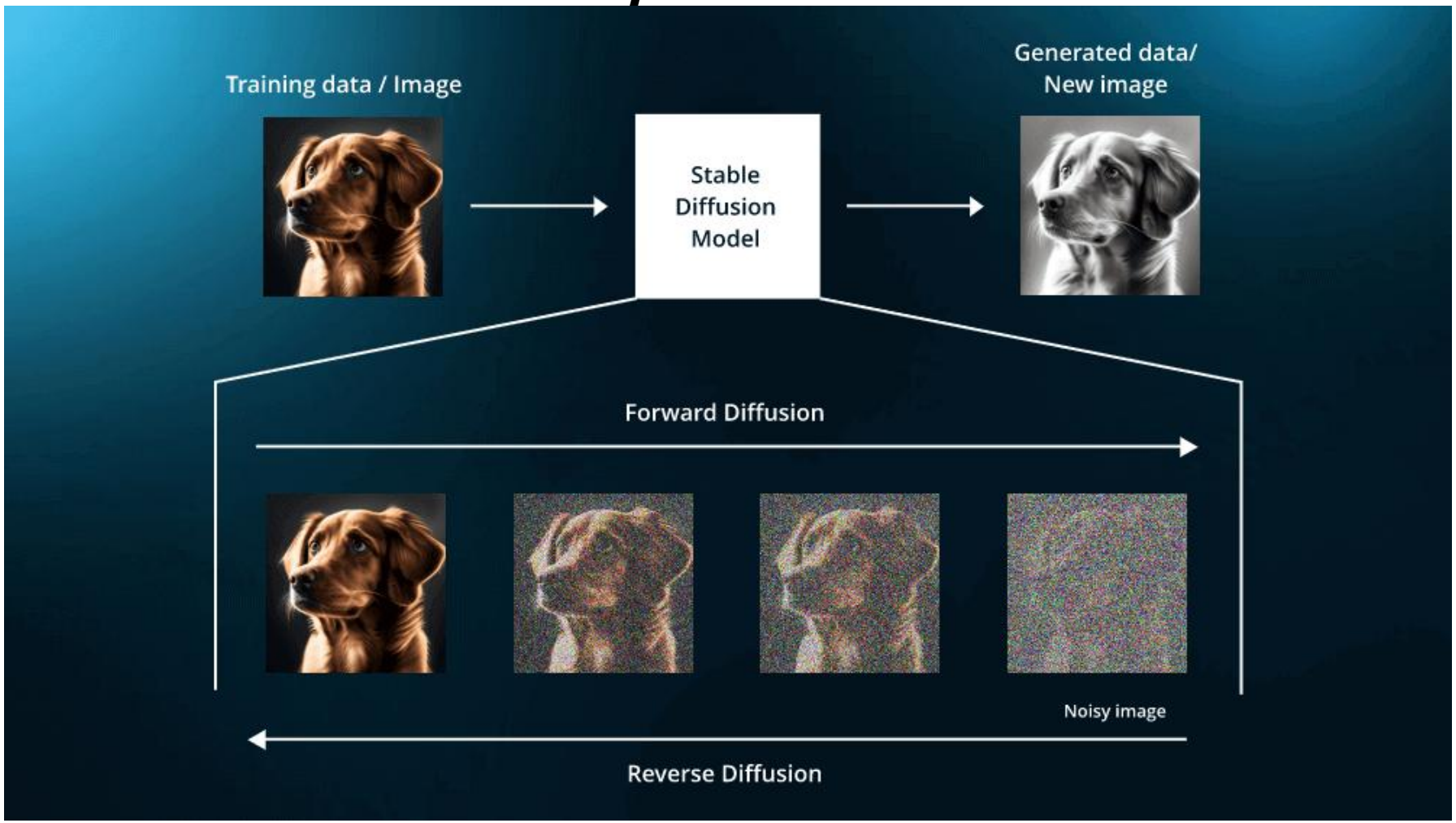
- **GAN'ler (Generative Adversarial Networks)**
  - Üretici modellerin popülerleşmesinde kilit rol oynadılar.
  - "İki ağın savaşı" fikrine dayanır:
    - Üretici (Generator): Sahte veri üretir.
    - Ayırt edici (Discriminator): Gerçek mi sahte mi karar vermeye çalışır.
- Bu rekabet sayesinde üretici ağ gittikçe daha gerçekçi örnekler üretmeyi öğrenir.
- Özellikle:
  - Foto-gerçekçi yüzler,
  - Sanatsal stil transferi (bir resmi Van Gogh tarzına dönüştürmek gibi),
  - Görüntü iyileştirme (çözünürlük artırma, gürültü giderme)alanlarında çok etkili olmuştur.

Neden Özellikle GAN ve Difüzyon Modelleri?

Difüzyon (Diffusion) modelleri

- Son yıllarda ön plana çıkan ve modern generatif sistemlerin çoğunun temelini oluşturan yapı.(DALL·E 3, Stable Diffusion vb. gibi sistemlerin arkasında difüzyon fikirleri yatar.)
- **Temel fikir:**
  - Bir görüntüyü al, üzerine adım adım gürültü ekleyerek tamamen boz.
  - Sonra bu gürültülü halden adım adım gürültüyü kaldırarak tekrar anlamlı bir görüntüye dönmeyi öğren.
- Bu süreç tersine çevrildiğinde:
  - Sadece gürültüden başlayıp yüksek kaliteli, detaylı görüntüler üretmek mümkün olur.
- **Avantajları:**
  - Çoğu zaman GAN'lere göre daha kararlı eğitim.
  - Çeşitlilik ve kalite bakımından çok güçlü sonuçlar.
  - Metin gibi koşullarla (prompt) yönlendirmeye çok uygun.

Difüzyon Modeli



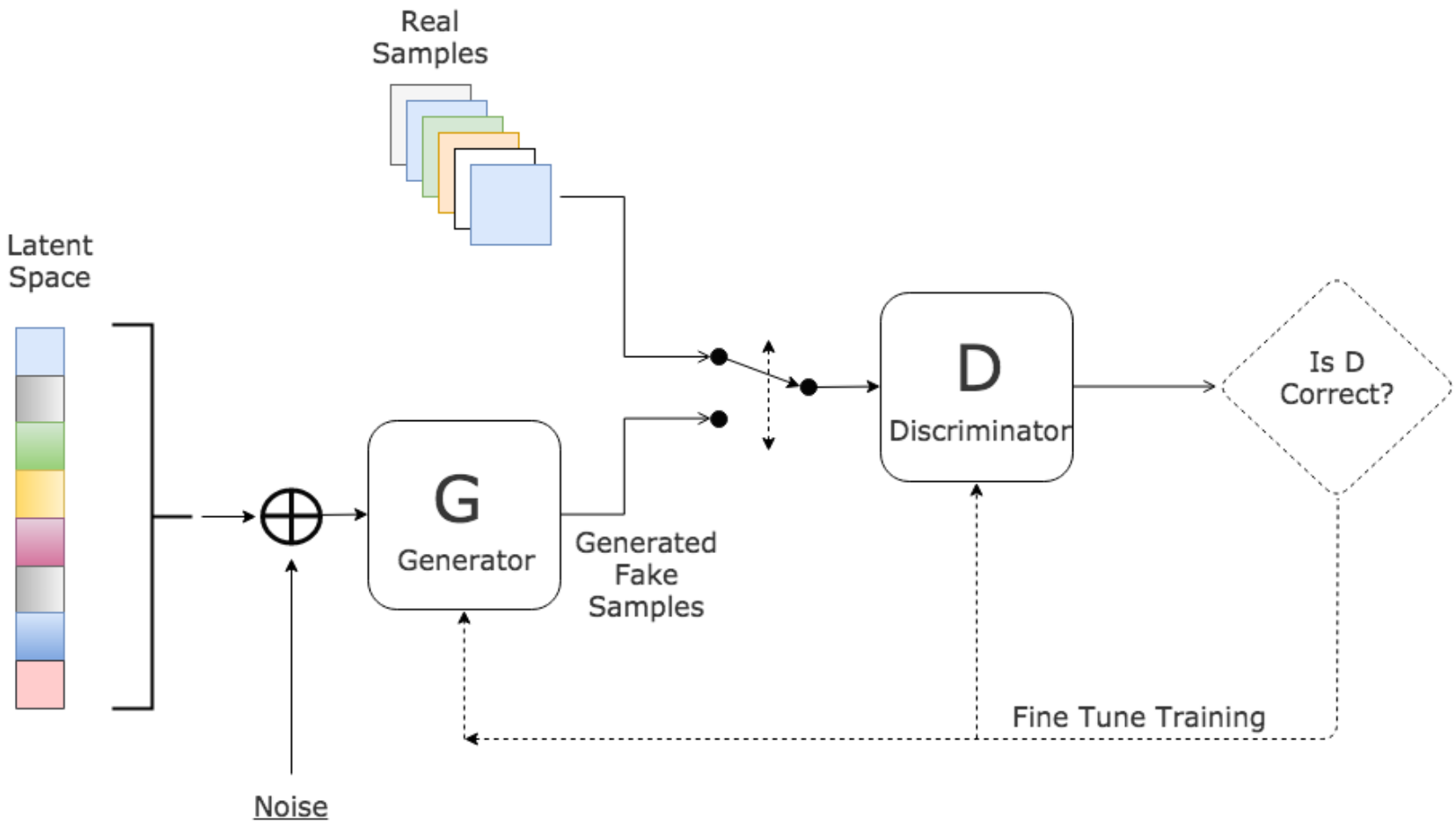
Neden Özellikle GAN ve Difüzyon Modelleri?

- GAN'ler, üretici model fikrini popülerleştiren ve kısa sürede göz alıcı örnekler sunan ilk büyük adım.

- Difüzyon modelleri, bu başarıyı bir adım öteye taşıyıp:
  - Daha kontrollü,
  - Daha kararlı,
  - Çok yüksek kaliteli üretim sağlayan yeni nesil yaklaşım.

- Bu nedenle "Üretici Modellerin Evrimi" dendiğinde, GAN → Difüzyon çizgisi temel bir tarihsel ve kavramsal eksen olarak görülebilir.

GAN


Real
Samples
Latent
Space
G
Generator
Generated
Fake
Samples
D
Discriminator
Is D
Correct?
Noise
Fine Tune Training

Temel Kavramlar - Üretici ve Ayırt Edici Modellerin Farkı

Üretici Modeller (Generative Models)

- **Amaç:** Eğitildiği veri kümesine benzeyen yeni örnekler üretmek.
- Ne yapar?
  - Elinde bir sürü kedi fotoğrafı var ⟶ model, kedilerin genel görünümünü öğrenir ⟶daha önce hiç var olmamış yeni kedi fotoğrafları üretebilir.
- Örnek alanlar:
  - Yeni yüzler üretme
  - Metinden resim oluşturma
  - Sentetik tıbbi görüntüler
  - Müzik, konuşma, metin üretimi

Üretici ve Ayırt Edici Modellerin Farkı

Ayırt Edici Modeller (Discriminative Models)

- **Amaç:** Verilen girdinin hangi sınıfa ait olduğunu tahmin etmek.
- Ne yapar?
  - "Bu resim kedi mi, köpek mi?"
  - "Bu yorum pozitif mi, negatif mi?"
  - "Bu mail spam mi, değil mi?"
- Örnek modeller:
  - Klasik CNN sınıflandırıcılar
  - Lojistik regresyon
  - SVM vb.

Üretici ve Ayırt Edici Modellerin Farkı

Özellik	Üretici Model	Ayırt Edici Model
Temel görev	Yeni veri üretmek	Veriyi sınıflandırmak / etiketlemek
Örnek soru	"Yeni bir yüz resmi üret."	"Bu yüz kim / hangi sınıf?"
Tipik kullanım	Görsel, ses, metin üretimi	Görüntü tanıma, spam tespiti vb.

Olasılıksal ve Deterministik Üretim

Olasılıksal Üretim (Probabilistic)

- **Tanım:** Aynı koşullarda bile model her çalıştırıldığında biraz farklı sonuç verebilir.
- **Neden?** Çünkü model bir yerlerde rastgelelik kullanır (örneğin "latent vektör" denen gizli girdi rastgele seçilir).
- **Örnek:**"Bana bir kedi resmi üret" dediğinde:
  - İlk çalıştırmada: siyah kısa tüylü bir kedi
  - İkinci çalıştırmada: beyaz uzun tüylü bir kedigibi çeşitli sonuçlar gelebilir.
- **Avantaj:** Aynı modelle çok sayıda farklı ve zengin örnek üretilebilir.
- **Nerede görürüz?** GAN'ler, difüzyon modelleri, VAE'ler ⟶ hepsi üretim sürecinde olasılıksal davranır.

Olasılıksal ve Deterministik Üretim

Deterministik Üretim (Deterministic)

- **Tanım:** Aynı girdi ⟶ her zaman aynı çıktı.
- **Örnek:** Bir sınıflandırıcıya aynı resmi ver:⟶ her zaman "kedi" diyorsa, bu deterministiktir.
- **Avantaj:** Sonuçlar tekrarlanabilir, test etmesi kolay.
- **Dezavantaj:** Eğer üretici bir model deterministik olsaydı,aynı koşullarda hep aynı resmi üretirdi ⟶ çeşitlilik kaybı olurdu.

Veri Dağılımı ve Latent Uzay Kavramları

Veri Dağılımı (Data Distribution)

- **Veri dağılımı nedir?**
  - Gerçek dünyadaki verilerin (yüzler, kediler, el yazıları, tıbbi görüntüler...) genel "nasıl göründüğü, ne sıklıkta ortaya çıktığı" bilgisidir.
- **Örnek:** Bir yüz veri setinde:
  - Bazı ten renkleri daha sık,
  - Bazı saç stilleri daha nadir,
  - Bazı pozlar (önden bakan yüz) daha yaygın.⟶ Bunların hepsi, veri dağılımının parçalarıdır.
- **Üretici modelin hedefi**
  - Gerçek veri kümesinin dağılımına benzer bir dağılım öğrenmek.
  - Böylece ürettiği örnekler gerçek veriyle uyumlu görünür.

Veri Dağılımı ve Latent Uzay Kavramları

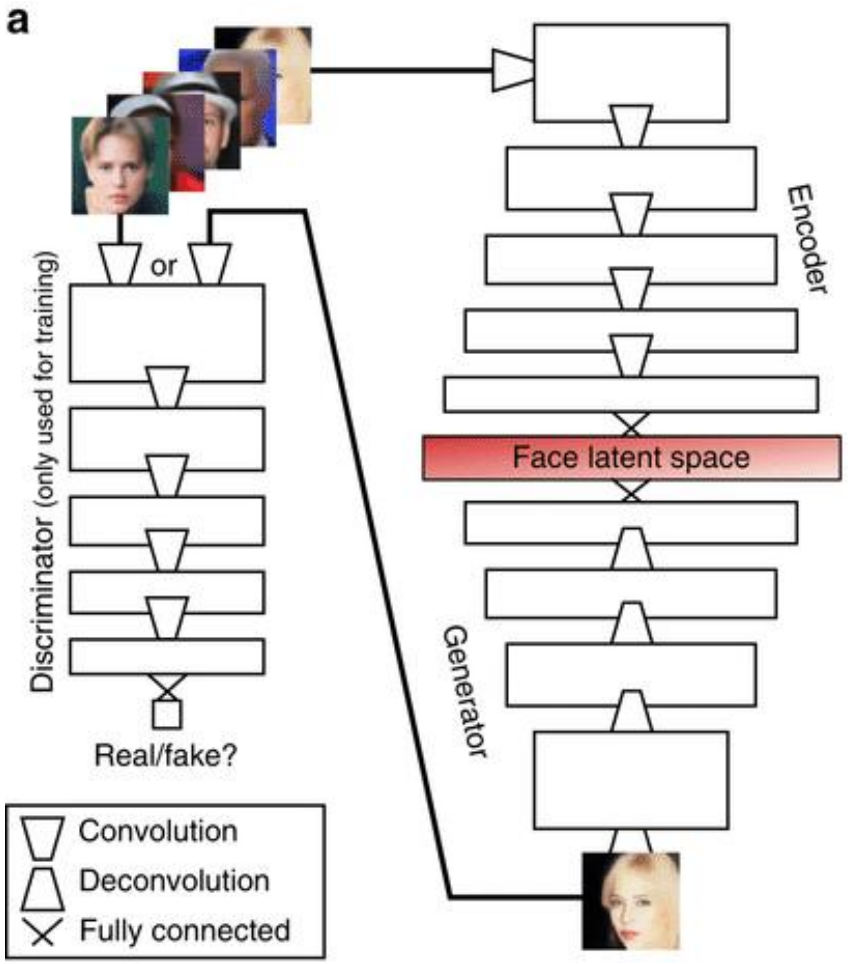
Latent (Gizli) Uzay

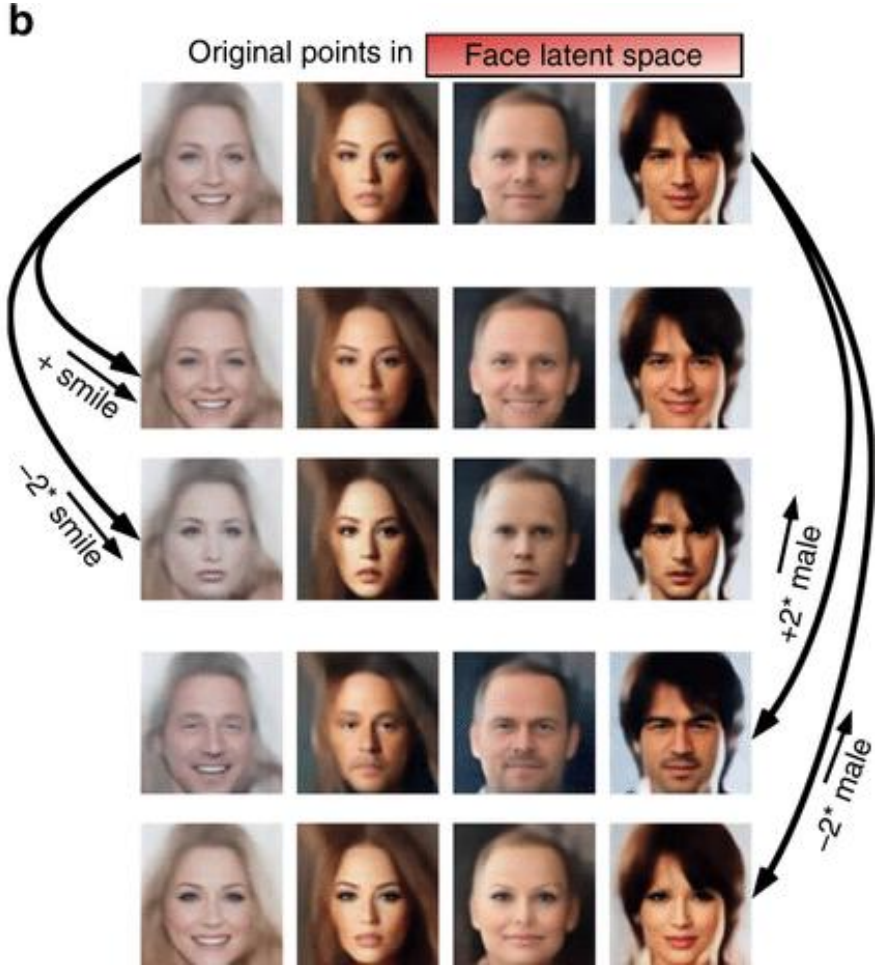
- **Latent uzay nedir?**
  - Modelin, veriyi temsil etmek için kullandığı soyut, gizli özellikler uzayı.
  - Biz doğrudan görmeyiz, ama model bu uzayda çalışır.
- **Elinde yüz fotoğrafları var.**
  - Model, her yüzü "gizli bir nokta" ile temsil ediyor olabilir:
  - Yaş, cinsiyet, saç uzunluğu, gülümseme miktarı, bakış yönü...gibi özellikler bu gizli noktada bir şekilde kodlanmıştır.
- **Latent uzayda yürümek**
  - Latent uzayda bir noktadan biraz sağa kayarsın ⟶ yüz biraz yaşlanır.
  - Başka bir yöne kayarsın ⟶ saç rengi değişir.
  - Arada bir noktaya gidersin ⟶ iki yüz arasında "ara bir yüz" üretmiş olursun.
- **Bu sebeple latent uzay**
  - Manipülasyon yapmak,
  - Stil değiştirmek,
  - Ara örnekler üretmek için çok kullanışlıdır.

Latent Uzay – Veri Uzayı İlişkisi

- **Veri uzayı:**
  - Gerçek verinin bulunduğu yer (örneğin 256×256 piksel görüntüler).
  - Boyut çok yüksektir.
- **Latent uzay:**
  - Daha düşük boyutlu (örneğin 64, 128, 512 boyutlu vektörler).
  - Verinin "özünü" taşıyan sıkıştırılmış bir temsil gibi düşünülebilir.
- Üretici modelin yaptığı şey:
- Latent uzaydaki bir noktayı alıp ⟶ veri uzayında bir örneğe dönüştürmek.
- Yani:
  - gizli vektör ⟶ yüz fotoğrafı
  - gizli vektör ⟶ manzara
  - gizli vektör ⟶ ses dalgası gibi.

Latent Uzay – Veri Uzayı İlişkisi





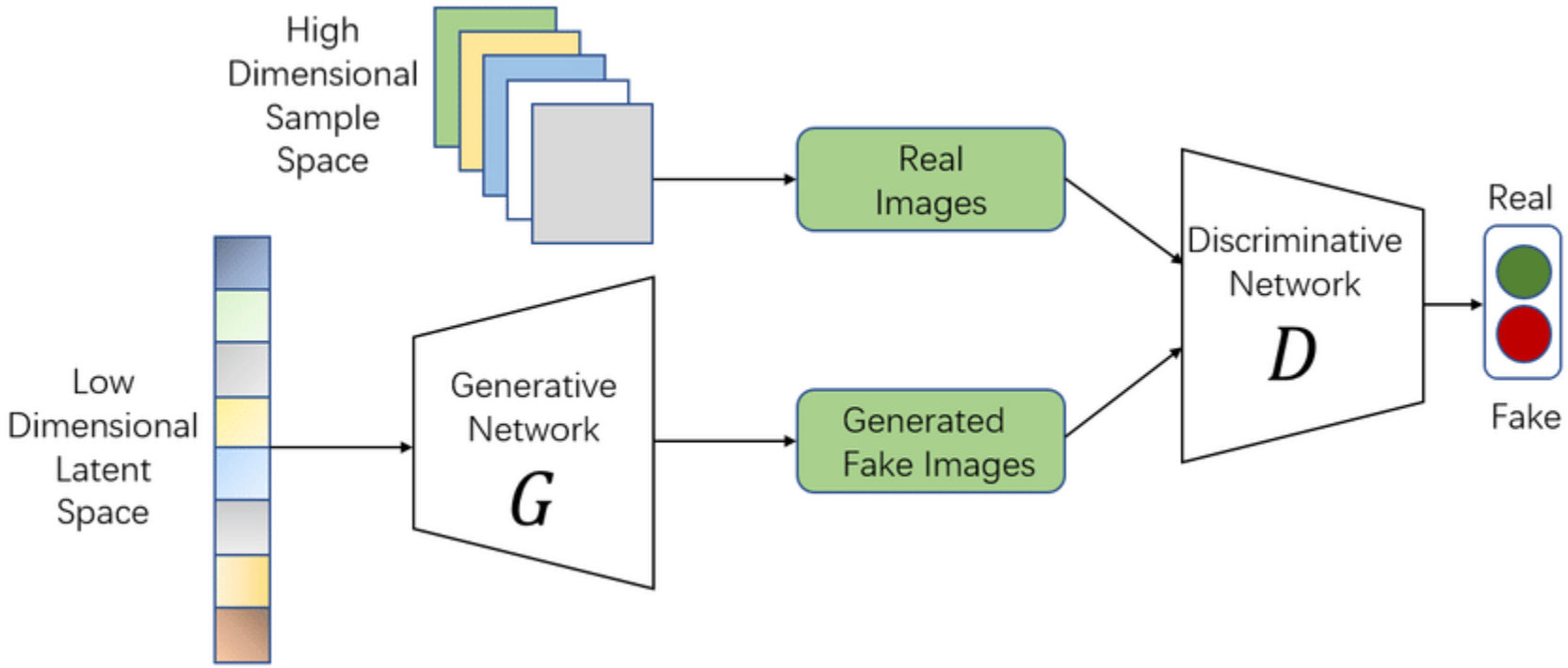
GAN Modelleri - GAN Nedir? (Ana Fikir)

- GAN, iki parçadan oluşan bir yapay zeka modeli:
  - Üretici (Generator) ⟶ sahte veri üretir.
  - Ayırt edici (Discriminator) ⟶ verilen verinin gerçek mi sahte mi olduğuna karar verir.
- Bunu şöyle düşünebilirsin:
  - Sahte para yapan biri (üretici)
  - Sahte parayı yakalamaya çalışan polis (ayırt edici)
- Sahte para yapan kişi gittikçe daha iyi sahte para yapar; polis de gittikçe daha iyi sahte parayı fark etmeyi öğrenir.
- Bu "karşılıklı yarış" sayesinde ikisi de gelişir.
- **GAN'in amacı:** Üretici o kadar iyi sahte veri üretsin ki, ayırt edici artık gerçek mi sahte mi ayıramasın.

GAN Modelleri - GAN Nedir? (Ana Fikir)

- Rastgele girdi (gürültü / latent vektör)⟶ sayıdan oluşan rastgele bir liste gibi düşün (örneğin 100 tane rastgele sayı).
- Bu rastgele girdi ⟶ Üreticiye (G) verilir.
- Üretici ⟶ bu girdiden sahte bir görüntü üretir (örneğin sahte bir yüz).
- Ayırt edici (D) ⟶ şu iki tür veri görür:
  - Gerçek görüntüler (veri setinden)
  - Üreticinin ürettiği sahte görüntüler
- Ayırt edici ⟶ her biri için "gerçek" ya da "sahte" tahmini yapar.
- Hatalara göre hem G hem D güncellenir (eğitilir).

GAN Modelleri - GAN Nedir? (Ana Fikir)



Eğitim Süreci: Adversarial Training (Karşılıklı Yarış)

- **Başlangıçta:**
  - **Üretici:** rezalet, saçma sapan şeyler üretir.
  - **Ayırt edici:** gerçek–sahteyi çok kolay ayırır.

Eğitim Süreci: Adversarial Training (Karşılıklı Yarış)

Eğitim döngüsü (çok kez tekrarlanır)

- Üretici, rastgele girdiden sahte görüntüler üretir.
- Ayırt edici:
  - Gerçek görüntüleri görür → "bunlar gerçek" demeyi öğrenir.
  - Sahte görüntüleri görür → "bunlar sahte" demeyi öğrenir.
- Sonra üretici güncellenir:
  - "Ayırt ediciyi kandıramadığım yerler neresi?"
  - "Sahte dedirttiğim hataları nasıl düzeltebilirim?"
- Böylece:
  - D → sahteyi yakalama konusunda gelişir.
  - G → daha gerçekçi sahte üretme konusunda gelişir.

Eğitim Süreci: Adversarial Training (Karşılıklı Yarış)

Amaç (ideal durumda)

- Üretici öyle seviyeye gelir ki:
    - Ayırt edici gerçek mi sahte mi %50–%50 tahmin etmeye başlar.
    - Yani artık "ayırt edemiyor".

GAN Çeşitlemeleri

- **DCGAN (Deep Convolutional GAN)**

- **Özellik:**
  - Tam bağlı katmanlar yerine evrişimli (convolutional) katmanlar kullanır.
  - Görüntü üretimi için daha uygundur.

- **Ne işe yarar?**
  - Daha düzgün, daha kararlı görüntü üretimi.
  - GAN'lerin görsel işlerde yaygınlaşmasına öncülük etmiştir.

GAN Çeşitlemeleri

- **StyleGAN**
- NVIDIA tarafından geliştirildi (çok duyduğumuz “gerçekçi sahte yüzler”in çoğu StyleGAN ile üretildi).
- **Özellik:**
  - Yüzün farklı özelliklerini kontrol etmeye izin verir:
    - Yaş
    - Cinsiyet
    - Gülümseme
    - Saç stili vb.
- **Sonuç:**
- Aşırı gerçekçi ve kontrol edilebilir yüz üretimi.

GAN Çeşitlemeleri

- **CycleGAN**
- **Kullanım alanı:**
  - Görüntüden görüntüye çeviri, üstelik eşlenmiş veri olmadan.
- **Örnekler:**
  - At ⟶ Zebra
  - Yaz ⟶ Kış
  - Gündüz ⟶ Gece
  - El çizimi ⟶ Fotoğraf benzeri görüntü
- Önemli nokta:"Şu giriş şu çıkışa denk gelir" diye birebir etiketli veri gerekmez. Sadece "atlar kümesi" ve "zebrâlar kümesi" yeterli.

Kodlama Zamanı

- 13_ganOrnek.py

Forward (Noising) Süreci – Bozma Aşaması

- Bu süreç eğitim sırasında kullanılır.

- **Ne yapılır?**
  - Gerçek bir görüntü alırsın.
  - Her adımda biraz daha gürültü eklersin:
    - **1.adım:** Hafif bozulmuş görüntü
    - **2.adım:** Daha bozuk...
    - **Son adım:** Neredeyse sadece "karıncalanma" (tam gürültü)
- **Özellikler**: Kuralları sabittir; model bu kısmı öğrenmez, sadece biz tasarlarız.
- Amaç, modelin sonradan tersini öğrenebileceği bir "bozma yolu" oluşturmak.

Reverse (Denoising) Süreci
Temizleme / Üretim Aşaması

- Asıl "sihir"in olduğu yer burası.

- **Modelin görevi:**
  - Gürültülü bir görüntü görünce,
  - "Bu bir önceki adımda nasıl görünmeliydi?" sorusuna cevap vermek.
- **Üretim sırasında:**
  - Tamamen rastgele gürültüyle başla.
  - Model bu gürültüyü biraz temizler ⟶ azıcık şekil belirmeye başlar.
  - Aynı şeyi defalarca tekrarlar ⟶ her adım biraz daha netleştirir.
  - Sonunda anlamlı bir görüntü oluşur (kedi, manzara, portre vs.).

Markov Zincirleri ve Sürecin Rolü

- **Difüzyon süreci adım adım ilerler:**
  - 1. adım → 2. adım → 3. adım → ... → N. Adım

- **"Markov" demek burada şu anlama gelir:**
  - Her adım, sadece bir önceki adımın sonucuna bakar.
  - 20. adımı hesaplarken "10 adım önce ne olmuştu?" ilgilenmeyiz;sadece 19. adımda ne olduğuna bakarız.

- **Neden önemli?**
  - Bu sayede süreci parçalara bölmek ve modelin her adımı ayrı ayrı öğrenmesini sağlamak daha kolay olur.

Score Matching ve Eğitim

- Difüzyon modeli, aslında şunu öğrenmeye çalışır:
  - "Bu gürültülü görüntüyü, daha gerçekçi olacak şekilde hangi yöne doğru düzeltmeliyim?"
- **Score matching**
  - Modele, verinin yoğunlaştığı bölgeleri (yani gerçekçi örneklerin bulunduğu "bölgeleri") öğretmeye yardımcı olur.
  - Model daha sonra gürültülü bir örnek görünce:
    - O örneği bu "gerçekçi bölgeler"e doğru kaydırmaya çalışır.
- Bunu şöyle hayal edebilirsin:
  - Dağınık bir noktalar bulutu (gürültü) var,
  - Sen "insan yüzlerinin toplandığı bölgeye doğru" noktaları çekmek istiyorsun.
  - Model her adımda bu çekme işini biraz yapıyor.

Latent Difüzyon Modelleri

- Şu ana kadar anlattığımız süreçler doğrudan piksel düzeyinde çalışıyor gibi düşünebilirsin
  - Yani her adımda 512×512 gibi büyük boyutlu görüntülerle uğraşmak zor.
- **Latent difüzyon ne yapar?**
  - Önce bir enkoder ile görüntüyü daha düşük boyutlu bir latent uzaya sıkıştırır
    - 512×512 piksel yerine, çok daha küçük bir temsil.
  - Difüzyon süreci (gürültü ekleme–temizleme) bu daha küçük uzayda çalışır.
  - En sonunda bir dekoder ile bu latent temsili tekrar yüksek çözünürlüklü görüntüye açar.
- **Avantajı:**
  - Çok daha hızlı ve hesaplaması ucuz hale gelir.
  - Yüksek çözünürlükte, detaylı görüntü üretmek mümkün olur.

GAN vs Difüzyon Modelleri

- **GAN'ler:**
  - Biraz daha "eski nesil" ama hâlâ çok güçlü üretici modeller.
  - Hızlı üretim, gerçekçi görüntüler, özellikle yüz ve stil konularında çok kullanıldılar.
- **Difüzyon modelleri:**
  - Yeni nesil üretici aile.
  - Özellikle metinden görüntü (DALL·E, Stable Diffusion vb.) tarafında baskın hale geldi.
- **Genel eğilim:**
  - "Hız ve pratiklik" gereken yerlerde → GAN
  - "Kalite, çeşitlilik ve kontrol" gereken yerlerde → Difüzyon

Performans Karşılaştırmaları - Görsel Kalite

- **GAN:**
  - Özellikle iyi eğitilmiş StyleGAN gibi modellerle, Çok gerçekçi insan yüzleri, Güzel stilize görüntüler üretebiliyor.
  - Ama bazı detaylarda, Arka planda garip artefaktlar, Tutarsız dokular görülebiliyor.

- **Difüzyon:**
  - Özellikle yüksek çözünürlükte çok detaylı ve temiz görüntüler üretebiliyor.
  - Işık, gölge, doku, küçük detaylar genelde daha başarılı.
  - Metinle yönlendirildiğinde (prompt) sahneye uygun detayları eklemede oldukça iyi.

Çeşitlilik (Diversity)

- **GAN**
  - En büyük problemlerden biri mode collapse: Veri setinde farklı örnekler olsa bile, model benzer birkaç örneğe "takılıp kalabiliyor".
  - Bu durumda çeşitlilik azalıyor.

- **Difüzyon**
  - Aynı metin girdisiyle bile farklı "seed" değerleri kullanarak geniş bir çeşitlilik elde edilebiliyor.
  - Gürültüden başlayan çok adımlı süreç,veri dağılımının farklı bölgelerini gezmeye daha elverişli.

Eğitim Kararlılığı

- **GAN**
  - İki ağın (G ve D) yarışı bazen dengesizleşiyor, D çok güçlüyse G öğrenemiyor,G çok güçlüyse D hiçbir şey öğrenemiyor.
  - Loss'lar dalgalanıyor, "neden bozuldu?" demek çok normal.
  - Eğitim hassas: mimari, öğrenme oranı, veri kümesi çok etkiliyor.

- **Difüzyon**
  - Eğitim süreci genelde daha kararlı: Tek büyük model var, "karşılıklı savaş" yok.
  - Uzun sürse de, kontrolü daha kolay;"bozuldu–dengesizleşti" problemi GAN kadar dramatik değil.

Kullanım Alanlarına Göre Tercih Sebepleri

Ne Zaman GAN?

- **Gerçek zamanlı / hızlı üretim gereken yerler:**
  - Oyun içi içerik üretimi,
  - Canlı video efektleri,
  - Anında geri bildirim gereken interaktif uygulamalar.
- **Belirli bir görev için özel olarak optimize edildiğinde:**
  - Yüz üretimi,
  - Stil transferi,
  - Süper çözünürlük (super-resolution) gibi spesifik problemler.
- **Kaynak kısıtlıysa (örneğin cep telefonu):**
  - Tek ileri geçiş (forward pass) ile çıktı alınabildiği için daha uygun.

Ne Zaman Difüzyon?

- **Görsel kalite ve detay önemliyse:**
  - Sanat, illüstrasyon, reklam görselleri,
  - Fotoğraf kalitesine yakın sahne üretimi.
- **Metinden görüntü / metinden video gibi karmaşık görevlerde:**
  - Prompt'la ince ayar vermek istiyorsan ("gece, yağmurlu, sinematik ışık" gibi).
- **Daha kontrollü düzenleme istiyorsan:**
  - Var olan bir görüntünün belli kısmını değiştirme (inpainting),
  - "Şu objeyi sil, buraya başka bir şey ekle" tarzı düzenlemeler.

Kodlama Zamanı

- **14_ganSahteYuz.py**

Kodlama Zamanı

- 15_difuzyonSahteYuz.py

Geleneksel Manipülasyonlar: Copy-Move ve Splicing

Görsel Manipülasyon (Image Tampering) – Genel Bakış

- Görsel manipülasyon nedir?
  - Dijital bir görüntünün içeriğinin kasıtlı olarak değiştirilmesi, ekleme/çıkarma yapılması veya sahte bilgi oluşturulması.

- **Amaçlar**
  - Sahte delil üretmek (ör. olay yerinde görünmeyen birini eklemek).
  - Nesneleri gizlemek (ör. bir kişiyi, objeyi, yazıyı kaldırmak).
  - Kitleleri yanıltmak, yanlış bilgi yaymak (ör. manipüle edilmiş haber görselleri).

Geleneksel Manipülasyonlar: Copy-Move ve Splicing

- **Neden tehlikeli?**
  - Yanlış bilgi yayılmasına neden olabilir.
  - Adli süreçlerde sahte delil üretilmesine yol açabilir.
  - Sosyal medyada itibar zedeleme, karalama kampanyaları gibi durumlara sebep olabilir.

- **Neden tespit zor?**
  - Modern düzenleme araçları (Photoshop, mobil uygulamalar vb.) çok güçlü.
  - Manipülasyonlar, çoğu zaman insan gözüyle fark edilmeyecek kadar iyi yapılabiliyor.

Geleneksel Manipülasyonlar: Copy-Move ve Splicing

- **Neden tehlikeli?**
  - Yanlış bilgi yayılmasına neden olabilir.
  - Adli süreçlerde sahte delil üretilmesine yol açabilir.
  - Sosyal medyada itibar zedeleme, karalama kampanyaları gibi durumlara sebep olabilir.

- **Neden tespit zor?**
  - Modern düzenleme araçları (Photoshop, mobil uygulamalar vb.) çok güçlü.
  - Manipülasyonlar, çoğu zaman insan gözüyle fark edilmeyecek kadar iyi yapılabiliyor.

Copy-Move Manipülasyonu

- **Copy-Move nedir?**
  - Aynı görüntünün içinden bir bölgeyi kopyalayıp yine aynı görüntü üzerine başka bir yere yapıştırma işlemidir.

- **Nerede kullanılır?**
  - İstenmeyen bir nesneyi gizlemek. Örnek: Fotoğraftaki çöp kutusunu, yanındaki duvarı çoğaltarak kapatmak.
  - Nesne çoğaltmak. Örnek: Kalabalık göstermek için aynı kişiyi/nesneyi birkaç kez kopyalayıp yapıştırmak.
  - Arka planı düzeltmek veya görüntüyü "daha düzenli" göstermek.
- Özet: Copy-Move, "içeriden kopyalayıp yine içeriye yapıştırma" işlemidir.

Copy-Move Manipülasyonu

- Copy-Move'da:
  - Kopyalanan bölge, aynı görüntüden alındığı için:
    - Renk tonu,
    - Doku,
    - Gürültü seviyesi,
    - Çözünürlük
  - hepsi birebir aynıdır.
- Klasik sahtecilik türlerinde (örneğin başka bir resimden parça ekleme):
  - Farklı kamera, farklı sıkıştırma, farklı ışık → tutarsızlık daha belirgin olabilir.
- Copy-Move'da ise:
  - Kullanılan kaynak ve hedef aynı olduğu için,
  - İnsan gözüyle de, basit istatistiklerle de fark etmek daha zor olur.
- Yani sorun şu: "Kopyalanan parça, görüntüye zaten ait; o yüzden yabancı durmuyor."

Copy-Move Tespiti – Temel Fikirler

a) Bölgesel benzerlik analizleri (blok bazlı karşılaştırmalar)

- Görüntü küçük parçalara (bloklara) bölünür.
  - Örneğin 8×8 veya 16×16 piksel bloklar.
- Her blok için:
  - Renk / doku gibi özellik vektörleri çıkarılır (özet bilgi).
- Bu bloklar birbirleriyle karşılaştırılır:
  - Aynı veya çok benzer özelliklere sahip bloklar tespit edilir.
- Eğer:
  - Benzer iki blok görüntüde farklı bölgelerde konumlanmışsa,
  - Aralarında belirli bir kayma (shift) varsa,⟶ Copy-Move şüphesi doğar.

Copy-Move Tespiti – Temel Fikirler

b) Dönüşüm, ölçek ve sıkıştırma etkilerine rağmen benzerlik arama

- Gerçek hayatta saldırgan:
  - Kopyaladığı bölgeyi hafif döndürebilir,
  - Küçük oranda büyültüp/küçültebilir,
  - JPEG sıkıştırma gibi işlemler uygulayabilir.

- Bu yüzden tespit yöntemleri:
  - Sadece "birebir aynı piksel"e bakmamalı,
  - Dönüşüm, döndürme, ölçekleme gibi küçük değişikliklere dayanıklı özellikler kullanmalı.

Copy-Move Tespiti – Temel Fikirler

c) Öznitelik (feature) tabanlı yöntemler: SIFT, PCA vb.

- **SIFT (Scale-Invariant Feature Transform):**
  - Görüntü üzerindeki "ayrıksı", dikkat çekici noktaların (keypoint) konumunu ve tanımlayıcılarını çıkarır.
  - Ölçek ve döndürmeye nispeten dayanıklıdır.
  - Aynı bölgenin kopyalanıp döndürülmüş/daraltılmış hali olsa bile benzer SIFT öznitelikleri bulunabilir.

- **PCA (Principal Component Analysis):**
  - Blokların veya yama (patch)'lerin özelliklerini daha düşük boyutlu bir uzaya sıkıştırmak için kullanılabilir.
  - Amaç, benzer blokları daha kolay ve hızlı kıyaslayabilmek.

Copy-Move Tespiti – Temel Fikirler

c) Öznitelik (feature) tabanlı yöntemler: SIFT, PCA vb.

- Bu tür yöntemlerde:
    - Farklı noktalardan çıkarılan öznitelikler karşılaştırılır,
    - Aynı/birbirine çok yakın olanlar bulunur,
    - Pozisyonlarına bakılarak "şüpheli eşleşmeler" tespit edilir.

Kodlama Zamanı

- **16_siftOrnek.py**

Splicing (Birleştirme) Manipülasyonu

- Splicing nedir?
  - İki veya daha fazla farklı görüntünün parçalarının alınıptek bir görüntüde birleştirilmesi işlemidir.
- Copy-Move'dan farkı:
  - Copy-Move ⟶ aynı görüntü içinde kopyala–yapıştır.
  - Splicing ⟶ farklı görüntülerden parçaları al, yeni bir kompozit görüntü oluştur.
- Örnekler:
  - Bir kişinin kafasını başka bir vücuda eklemek.
  - Olay yerinde hiç bulunmamış bir kişiyi, başka bir fotoğraftan kesip sahneye yerleştirmek.
  - Gökyüzüne başka bir fotoğraftan alınmış patlama, duman, UFO vb. eklemek.
  - Farklı zamanlarda çekilmiş fotoğrafları tek karede birleşmiş gibi göstermek.
- Özet: Splicing, "farklı kaynaklardan parçaları al, tek kareymiş gibi göster" sahteciliğidir.

Splicing (Birleştirme) Manipülasyonu

- Tehlike:
  - Her bir parça zaten "gerçek bir fotoğraftan" geldiği için, İnsan gözüne oldukça inandırıcı gelebilir.
  - Tek kareymiş gibi sunulduğunda, O kişi gerçekten oradaymış,O olay gerçekten öyle yaşanmış gibi yanlış bir algı oluşturabilir.
- Copy-Move'a göre ek zorluklar:
  - Farklı cihazlar / objektifler / çözünürlükler kullanılmış olabilir.
  - Farklı ışık koşulları, farklı JPEG sıkıştırma seviyeleri, farklı gürültü yapıları işin içine girer.
- Aynı zamanda bu farklılıklar, Tespit için de ipucu olabilir(çünkü "doğal olmayan" tutarsızlıklar ortaya çıkar).

Splicing Tespitinde Kullanılan Temel İpuçları

- Farklı görüntülerden gelen parçaların:
  - Kenarlarında,
  - Birleşim bölgelerinde çoğu zaman "bir şeyler hafifçe sırıtabilir".
- Dikkat edilen noktalar:
  - Kenar yumuşaklığı/sertliği (blur seviyesi aynı mı?).
  - Arka plan dokusu, çizgiler, izler devamlı mı?
  - Renk geçişleri doğal mı, yoksa keskin ve yapay mı?
- Örnek:
  - Arkada gökyüzü hafif grenli ve bulanık,
  - Eklenen uçak çok keskin ve pırıl pırıl görünüyorsa → şüpheli.

Metadata Nedir?

- **Metadata (üstveri):**
  - Bir veri hakkında "verinin kendisi olmayan ama onu tanımlayan, açıklayan ek bilgiler"dir.

- Yani:
  - Fotoğraf = veri
  - Fotoğrafın ne zaman çekildiği, hangi cihazla çekildiği, çözünürlüğü, konumu vb. = metadata

- Kısaca: "Veri hakkında veri."

Günlük Hayattan Örnekler

- **Bir fotoğrafın metadata'sı:**
  - Çekim tarihi ve saati
  - Kamera/telefon modeli
  - En-boy çözünürlüğü
  - Pozlama süresi, diyafram, ISO gibi teknik bilgiler
  - GPS konumu (enlem, boylam)
- **Bir müzik dosyasının metadata'sı (ID3 tag):**
  - Şarkı adı
  - Sanatçı
  - Albüm adı
  - Tür (genre)
  - Çıkış yılı
- **Bir Word/PDF belgesinin metadata'sı:**
  - Belge başlığı
  - Yazar adı
  - Oluşturma/değiştirme tarihleri
  - Kullanılan yazılım (ör. "Microsoft Word 2019")

Metadata Nedir?

- Metadata çoğu zaman:
  - Dosyanın içinde gömülü durur,Ekranda doğrudan görünmez.

- Fotoğrafı normal bir görüntüleyici ile açtığında:
  - Sadece görüntüyü görürsün, Ama dosyanın içinde tarihler, konum, cihaz bilgisi vb. saklanıyor olabilir.

- Bu yüzden:
  - Gizlilik / güvenlik açısından önemli,
  - Adli bilişim / sahtecilik analizi açısından da çok değerlidir.

Metadata Türleri

- **Teknik metadata:**
  - Dosya boyutu, çözünürlük, format (JPEG, PNG, MP4), sıkıştırma oranı vb.
- **İçerik metadata'sı:**
  - Başlık, açıklama, etiketler (tag), konu, kategori vb.
- **Yönetimsel (idari) metadata:**
  - Oluşturan kişi, kurum, telif bilgisi, sürüm, değişiklik tarihleri vb.
- **Jeo-metadata (konum):**
  - GPS koordinatları, çekim noktası, harita bilgisi vb. (özellikle fotoğraflarda)

Sahtecilik ve Adli Analiz Açısından Metadata

- **Bir fotoğrafın metadata'sı şunlar için ipucu verebilir:**
  - Gerçekten ne zaman çekilmiş?
  - Hangi cihazla çekilmiş?
  - Çekildiği konum neresi?
  - Dosya üzerinde sonradan hangi yazılımla işlem yapılmış?

- **Örneğin:**
  - "Bu fotoğraf olay anında çekildi" iddiası varken,
  - Metadata'da tarih farklı bir gün / yıl ise,
  - Ya da cihaz türü / yazılım bilgisi tutarsızsa ⟶ şüphe oluşur.

EXIF Nedir

- EXIF: Exchangeable Image File Format
- Dijital fotoğraflar ve bazı görüntü dosyaları içinde saklanan standart bir metadata (üstveri) yapısıdır.

- Özellikle:
  - Dijital fotoğraf makineleri
  - Akıllı telefon kameraları
- tarafından çekilen fotoğraflarda çok yaygın olarak kullanılır.

- Özet: EXIF, fotoğrafın nasıl, ne zaman, hangi ayarlarla çekildiğini anlatan bilgi paketidir.

EXIF Nedir

- **EXIF verisi:**
  - Genellikle JPEG, TIFF gibi resim dosyalarının içine gömülü olarak saklanır.
  - Dosyanın "görünmeyen" kısmındadır; yani görüntüye baktığında fark etmezsin.
- Bir fotoğrafı bilgisayarda "özellikler / detaylar" menüsünden veya özel araçlarla görüntülediğinde bu bilgilere erişebilirsin.

📷 Kamera ve Çekim Bilgileri

- Kamera/telefon **marka ve model**i
  - Örn: `Canon EOS 80D`, `iPhone 13`
- Pozlama süresi (shutter speed)
  - Örn: `1/125 s`
- Diyafram değeri (f-number)
  - Örn: `f/2.8`
- ISO hassasiyeti
  - Örn: `ISO 400`
- Odak uzaklığı (focal length)
  - Örn: `35 mm`

🗓️ Zaman Bilgisi

- Çekim zamanı:
  - **Tarih ve saat** (örn. `2025:03:10 14:32:05`)
- Bazı cihazlarda:
  - Son düzenleme zamanı
  - Zaman dilimi bilgisi

📍 Konum Bilgisi (GPS)

- Cihazda GPS açıksa:
  - Enlem (latitude)
  - Boylam (longitude)
  - Yükseklik (altitude)
- Bu bilgilerle:
  - Fotoğrafın harita üzerinde çekildiği nokta bulunabilir.

💻 Yazılım ve İşleme Bilgileri

- Fotoğrafın işlendiği / kaydedildiği yazılım:
  - Örn: `Adobe Photoshop`, `Snapseed`, `GIMP`
- Dosyanın son kaydedildiği tarih
- Bazı durumlarda kullanılan uygulama sürümü

EXIF ve Metadata İlişkisi

- EXIF, aslında metadata türlerinden sadece biridir.

- Yani:
  - Tüm metadata = daha genel kavram
  - EXIF = özellikle görüntü dosyaları (fotoğraflar) için kullanılan özel bir metadata formatı.

- Fotoğraf özelinde konuştuğumuzda:
  - "Metadata'ya bak" → çoğu zaman "EXIF verisine bak" anlamına gelir (ama ikisi birebir aynı şey değil).

Neden EXIF Özel Olarak Adlandırılıyor?

- Çünkü:
  - Birçok cihaz ve yazılım, EXIF standardını ortak bir dil gibi kullanıyor.
  - Adli bilişim, gazetecilik, araştırma vb. alanlarda:
    - "EXIF'ine bak" dendiğinde,
    - Fotoğrafın içindeki teknik ve tarih–konum bilgilerine bakılması kastedilir.
- EXIF, sahtecilik analizinde de sık sık başvurduğumuz ilk bilgi kaynağıdır.

Metadata / EXIF Neden Önemli

- Metadata/EXIF, **verinin bağlamını** verir:
  - *Ne zaman üretildi?*
  - *Nerede üretildi?*
  - *Hangi cihaz/yazılımla üretildi?*
- Sadece görüntüye bakarak göremeyeceğin bilgileri sağlar.
- Bu yüzden:
  - **Doğrulama** (verification),
  - **Takip / izleme** (traceability),
  - **Adli analiz** (forensics),
  - **Gizlilik / güvenlik**

    açısından kritik rol oynar.

Adli Bilişim ve Sahtecilik Tespiti Açısından

- Bir fotoğrafın EXIF bilgileri ile:
  - **İddia edilen tarih** tutuyor mu?
  - **İddia edilen konum** tutuyor mu?
  - **Aynı cihazdan çıktığı** söylenen fotoğraflar gerçekten aynı cihazdan mı?
- Örnek:
  - "Bu fotoğraf olay anında çekildi" deniyor ama:
    - EXIF → tarih 2 hafta öncesini gösteriyor.
    - Ya da EXIF → tamamen farklı bir ülkede çekildiği görülüyor.
  - Bu durumda **ifadeyle fotoğraf çelişiyor** → sahtecilik şüphesi artar.
- Birden fazla fotoğraf:
  - Aynı kamera ile çekilmişse sensör/gürültü izi + EXIF karşılaştırılarak doğrulama yapılabilir.

Kullanıcı Gizliliği / Güvenlik Açısından

- EXIF içinde yer alan **GPS koordinatları**, kişisel güvenlik açısından risklidir:
  - Evde çekilen bir fotoğraf → EXIF'ten ev adresinin yaklaşık konumu bulunabilir.
  - Çocukların, özel alanların fotoğrafları paylaşıldığında konumun da sızmasına yol açabilir.
- Bu yüzden:
  - Pek çok platform (WhatsApp, Twitter vb.) paylaşılan görsellerden EXIF'i siliyor.
  - Kullanıcıların da hassas fotoğrafları paylaşmadan önce EXIF'i temizlemesi önerilir.
- Ayrıca:
  - Cihaz modeli, yazılım bilgisi gibi detaylar:
    - Hedefli saldırılar veya sosyal mühendislik için de kullanılabilir.

Dijital Arşivleme ve Yönetim Açısından

- Büyük veri arşivlerinde:
  - Fotoğrafları **tarihe, konuma, cihaza** göre ayırmak için metadata/EXIF kullanılır.
- Örneğin:
  - Bir kurumun medya arşivi:
    - "2022 yılı, İstanbul'da çekilmiş, Canon kamera ile çekilmiş fotoğraflar" gibi filtrelemeleri metadata sayesinde yapabilir.
- Bu:
  - Arama,
  - Kataloglama,
  - Yedekleme ve saklama süreçlerinde büyük kolaylık sağlar.

Sahtecilik Tespitinde EXIF İpuçları

EXIF tek başına **kesin kanıt** değildir, ama çok faydalı ipuçları verir:

- **Tutarsız tarih/saat:**
  - Olay saati ile EXIF saati uyuşmuyor.
- **Tutarsız konum:**
  - Kişinin o sırada orada olması iddiasıyla çelişen GPS konumu.
- **Şüpheli yazılım bilgisi:**
  - "Bu fotoğraf direkt telefondan geldi" denirken,
  - EXIF'te ağır düzenleme yazılımları görünüyor.
- **Birden fazla fotoğraf arasındaki tutarsızlıklar:**
  - Aynı olay için:
    - Birinin EXIF'inde farklı saat, diğerinde farklı cihaz,
    - Biri GPS'li, diğeri tamamen eksik,
    - Ya da mantıksız sıralama.

> Önemli: Hatalı saat, yanlış ayarlanmış tarih, EXIF silme gibi durumlar da mümkündür.
> Bu nedenle EXIF, **"güçlü bir destekleyici bilgi"**, ama tek başına "mahkeme kararı" değil.

EXIF Verisinin Sınırlamaları - EXIF Her Zaman Yoktur

- Her görüntü dosyasında EXIF bulunmaz.
- EXIF'in olmayabileceği durumlar:
  - **Ekran görüntüleri (screenshot)**:
    - Genellikle EXIF içermez.
  - **Bazı sosyal medya / mesajlaşma uygulamaları**:
    - Fotoğrafı yüklerken EXIF'i otomatik olarak silebilir (gizlilik için).
  - **Bazı düzenleme / sıkıştırma işlemleri**:
    - Fotoğraf yeniden kaydedilirken EXIF bilgisi atılabilir.
- Sonuç:
  - "EXIF yok → kesin sahte" demek yanlıştır.
  - Sadece "bu fotoğraf üzerinde ara işlem yapılmış olabilir" gibi bir ihtimal doğar.

EXIF Kolayca Değiştirilebilir

- EXIF, dosyanın içinde bulunan **düzenlenebilir bir veri**dir.
- İnternette:
  - EXIF düzenleme araçları, masaüstü programları, mobil uygulamalar mevcuttur.
  - Örneğin:
    - Çekim tarihini değiştirebilirsin,
    - Konum ekleyebilirsin veya silebilirsin,
    - Kamera modelini farklı gösterebilirsin.
- Sonuç:
  - EXIF'te gördüğün bilgi **%100 güvenilir "gerçek" olmak zorunda değildir**.
  - Özellikle manipülasyon niyeti olan biri, EXIF'i kendi hikâyesine uygun hale getirebilir.

Cihaz Saat/Tarih Ayarları Hatalı Olabilir

- EXIF'teki tarih–saat bilgisi:
  - Cihazın **sistem saatine** bağlıdır.
- Eğer:
  - Kullanıcının cihazı yanlış tarih/saatte ayarlıysa,
  - Saat otomatik güncellenmiyorsa,
  - Kullanıcı bilinçli olarak zamanı geriye/ileriye almışsa,
- EXIF tarih–saat:
  - Gerçek çekim zamanını yansıtmayabilir.
- Sonuç:
  - "EXIF'teki tarih–saat, gerçek zaman" → doğru olmayabilir.
  - Yine sadece **yardımcı bir ipucu** olarak değerlendirilmelidir.

EXIF Bilgisi Eksik veya Kısıtlı Olabilir

- Her cihaz / uygulama:
  - EXIF'in tüm alanlarını doldurmayabilir.
- Örneğin:
  - Bazı telefonlarda GPS kapalıysa:
    - Konum alanları tamamen boş kalır.
  - Bazı basit kameralar:
    - Sadece temel teknik bilgileri (diyafram, ISO, model) yazar,
    - "Software" veya detaylı metadata alanlarını doldurmaz.
- Sonuç:
  - "Bu EXIF'te GPS yok → demek ki masaüstünde manipüle edilmiş" gibi net bir yargıya *varılamaz*.

Sosyal Medya, Mesajlaşma ve EXIF Temizliği

- Birçok platform:
  - Yüklenen fotoğrafların EXIF'ini **otomatik olarak temizler**:
    - Mahremiyet,
    - Veri tasarrufu,
    - Standartlaştırma gibi sebeplerle.
- Örneğin:
  - WhatsApp ile bir fotoğraf gönderdiğinde:
    - Çoğu zaman EXIF bilgisi karşı tarafa gitmez.
  - Twitter / Instagram'a yüklenen fotoğraflar da genelde EXIF'siz hale gelir.
- Bu durum:
  - Adli analiz açısından güçlük yaratabilir (EXIF kaybolur),
  - Ama kullanıcı gizliliği açısından çoğu zaman olumludur.

Kodlama Zamanı

- 17_exifOrnek.py