

Glieler Team :

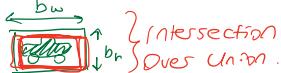
YOLOV4 Kullanılmış. → nesne algılama için, Darknet ve YOLOV4 için Visual Studio kullanılmış.

Compile için "Cmake" kullanılmış.

Veri seti : Open Images Dataset v6 *

Calisma prensibi : Residual blocks, Sınırlayıcı kutu regresyonu, Intersection Over Union

→ MID_s → nesneleri sınıfta sınıflandırılmış.



→ Tek GPU üzerinde görüntü

Yardimcilar : OpenCV, Nvidia, CUDA / Yapay zeka eğitimi : Darknet

Veri tesisleri : Veri tesisleri

+ cmake (Linux'da çalıştırıyor)

ADYU AI Team :

Faster R-CNN, YOLOV4, DyHead hibrit inceleme *

YOLOV4 insan görüntüsünde doğruluk oranı düşük dektir.

Tensorflow Object Detection API *

Python kullanılmış ✓

Algoritmalar : SSD, Yolo

* Uçan ambulans ve uçan araba gibi ile veri denetleme.

→ tek erişim noktası, sınırlayıcı kutular ve sınıf olasılıkları

+ Araçların belirleme acısı hassasiyeti
üzerinde çalışma.

Görüntü ve donanım Amazon-google GPU'nın
Cloud servisinde eğitilmiştir *

UAV görüntülerini veri setlerini, drone görüntülerini
kullanma ✓ (UAV datasetleri, harmanlama)

Dyhead ile R-CNN birleştirme fikri *

↳ Dynamic Head : Unifying Object Detection

+ Heads with Attention.

+ LocNet : Improving Localization Accuracy for
Object Detection. (IoU doğruluk istismar)

+ Wei JIANG : Improve Object Detection by
Data Enhancement based on GAN (görüntü
kalitesi) (istismar)

+ Vehicle and Person Tracking in
UAV Videos

MERGEN TEAM :

Yolcu adaklere ek olarak +1 veri seti isteme seti : YOLOv4-tiny

Veri etiketleme sistemi : Labelling

* Yapay sinir katmanlarından oluşmaktadır.

+ CSPDarknet53 → kategorizasyon

+ Backbone için → 3 (görsellerin önemli özelliklerini içeren)

+ Neck → SPP (mekansal piksel ortalaması), Block ve DANet (blok toplama)

+ Head → YOLOv3 → veri al ve tanıma sistemi

+ Head → YOLOv3 → veri al ve tanıma sistemi

* İnsan gölgelerini insan olarak algılayabiliyor. → Nesneye kırma yöntemi 777

* Hareket kesintileri ↴

* Görüntüye gürültü ekleyip onu temizleme yöntemiyle test 777

* IHA'dan çekilen daha fazla veri 111

Nesne tespiti : YOLO Algoritmaları
(YOLOv4)

→ genel zamanlı nesne algı-
lamada en yüksek model

FEZA AI TEAM :

YOLOv5 kullanılmış. → PyTorch hub

Rockstar Advanced Game Engine ile 3D yapay görsel gerçekçi (daha gerçekçi)

* Küçük nesneler için SAHİ

Algoritmaları

YOLOUS TEAM :

Unreal Engine 5 oyun motoru ile sahnede neme !!

→ gerçeğe yakın görseller.

+ YOLOv5 kullanımı ✓

1 TH GPU ✓

Görsel kalite iyileştirmesi için : GMC filtresi → Freaky Details → kötü hava koşulları için ✓

Kullanılan veri setleri : Teknofest vs, Vrdrone 2019 vs, VEDATI vs, Senteftik vs ✓

BIG BRAI NET → En iyi performansı Yolo v5/6 vermiş.
Yolov5, VisDrone, COCO

Müzeyyen Han : (VAILD)

Veri setleri: Roundabout, Vehicle Aerial Imaging from Drone, UAVDT, LSM

Nonmax suppression algoritması ile gereksiz sınırlayıcı kutularдан kurtulmuş *

* U-Net → maske görüntüsü ebe etme için.

* İmzı durumu başarılı belirtilmeli → durum "-1" de olmalı
+ EN düşük tespit başarısı insan sınıfında, panelvan ve yük taşıma kapasitesine sahip erkekler
için hatalı sonucu olasılığı, gölgelerde sıklık.

FEZA :

TPH-Yolov5 ve VisDrone ✓

Algolar MMDetection ile oluştur.

Backbone için → Visual Attention Network, Conv Next, Swin

Neck → Bidirectional feature Pyramid, (FPN)

Veri setleri → UAV, VisDrone, Aerial Dataset, AF0

PADIM algoritması → Piste işsiz durumu için

Örn/ConvNext + PANet + T-head

Nesne algosu

(TPH-Yolo v5)+(WBF) ⚡ data degru
Sonuç veriyor.

data başarılı

SOFTWARE :

Standford drone, PKLot, Cars Overhead with Context, Visdrone 2020

Veri setleri: Standford drone, PKLot, Cars Overhead with Context, Visdrone 2020
Scaled - Yolov4 algoritması kullanılmış.

Kesilen karelerin benzerlik oranı için → feature matching, histogram karşılaştırma istemleri ✓

Eşsiz veri setinin eğitim-test verileri olarak bölünmesi için → K-fold Cross Validation *

Python, numpy array yapıları kullanılmış, Numba kütüphanesi ile hızlandırılmış.

Python, numpy array yapıları kullanılmış, Numba kütüphanesi ile hızlandırılmış.

Pencere, konteyner, cell, direkt sıklık ölçüyor.

Tugra, Arxiv, Pesmod, Detrac, UAVDT, Arxiv, Pesmod, Tugra

WENN :

Veri setleri: Vedai, Dota, VisDrone, Bdd100k, VOC, COCO, Dir, Detrac, UAVDT, Arxiv, Pesmod, Tugra

Cycle GAN → hava koşulları için ✓

Algolar: Yolov5, Yolov7, Yolox, SATL ✓

ÜÇA ile sentetik veri üretimi ✓

ÜÇA ile sentetik veri üretimi → kaç gerçekde doğru ölçüyor.

Precision değeri ile keskinlik → kaç gerçekde doğru ölçüyor.

ZEKAI :

TPH-Yolov5 ✓

VisDrone veri seti ✓

ResNet2 numarisi ✓

Veri artırma için → Mixup ve mozaik algoritmalar! ***

Neck → SFA, AFF ✓

Omurgada DarkNet53 kullanılmış.

Backbone → DYTTH ✓

SATL Algoritması + ✓

GÖKKURT - YZ:

YOLOV5 ✓ * + Veri setleri: VisDrone, standford, Aerial Semantic, ERA, UAVDT ✓

Çift Modeli yap' gelistirmek için YOLOR

Hiper-parametre optimizasyon calismaları

Renk gecis sistemi → gece ve kötü hava da color saturation ile ayarlama

Renk gecis sistemi → gece ve kötü hava da color saturation ile ayarlama

DEEP TEACHING:

Kullanicilar mimari: DETR-DCS, Deformable DETR, u çok transformer mimarisini deneyimler mi.

Veri setleri: VisDrone, Standford, UAVDT

Algo: Two stage deformable DETR → $\text{pix} \times 2 \text{ seq}$, DINO, Query Det *

↳ kendi veri setleri