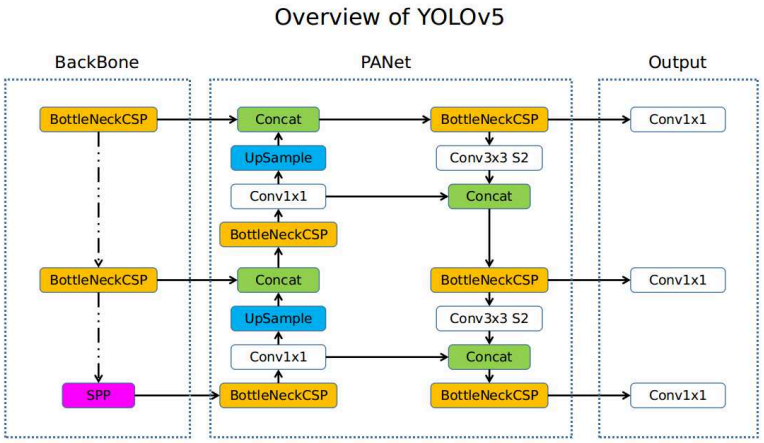
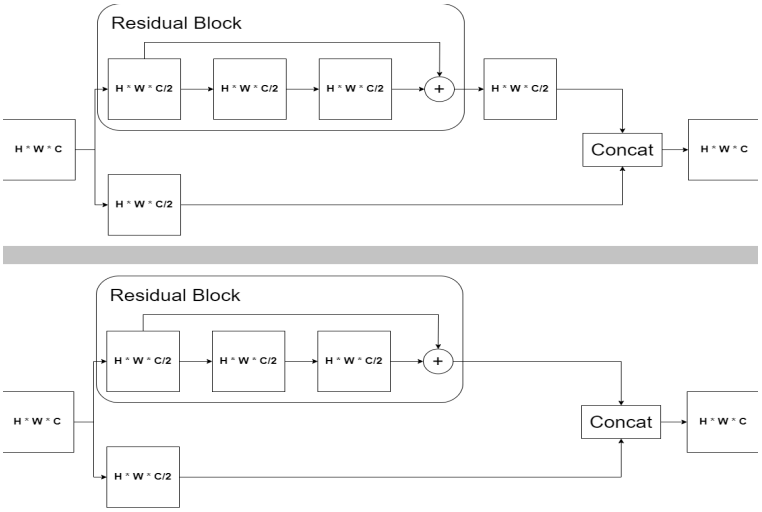
1. YOLOv5

1.1. Kiến trúc:



1.2. Backbone:

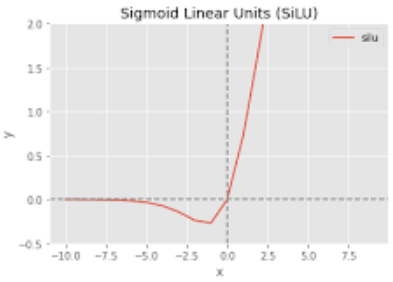
C3 module: YOLOv5 cải tiến CSPResBlock của YOLOv4 thành một module mới, ít hơn một lớp Convolution gọi là C3 module



1.3. Activation function và Loss:

- Activation func: SiLU (Sigmoid-weighted Linear Unit)

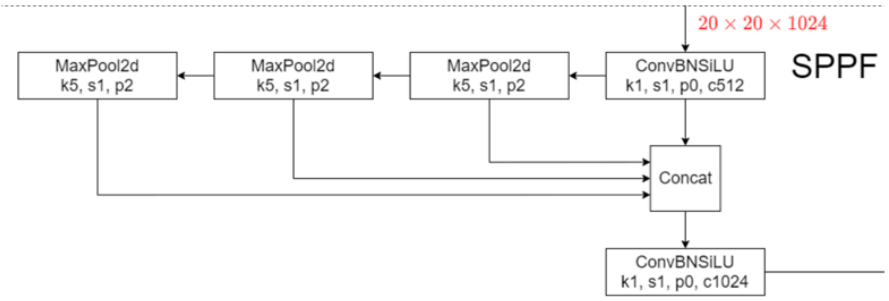
$$SiLU(x) = x\left(\frac{1}{1+e^{-x}}\right)$$



- Loss: Binary Cross Entropy và Logit Loss Function

1.4. Neck:

- YOLOv5 áp dụng một module giống với SPP, nhưng nhanh hơn gấp đôi và gọi đó là SPP - Fast (SPPF). Thay vì sử dụng MaxPooling song song như trong SPP, SPPF của YOLOv5 sử dụng MaxPooling tuần tự. Hơn nữa, kernel size trong MaxPooling của SPPF toàn bộ là 5 thay vì là [5,9,13] như SPP của YOLO v4



Hình 12. Kiến trúc của module SPPF

1.5. Xử lý data:

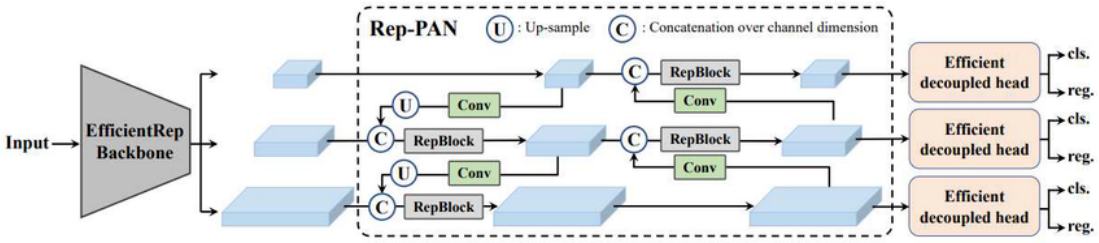
- Các kỹ thuật Data Augmentation được áp dụng trong YOLOv5 bao gồm:
  - Mosaic Augmentation
  - Copy-paste Augmentation
  - Random Affine transform
  - MixUp Augmentation
  - Thay đổi về màu sắc cũng như là Random Flip của Albumentations

1.6. Thay đổi so với bản trước đó:

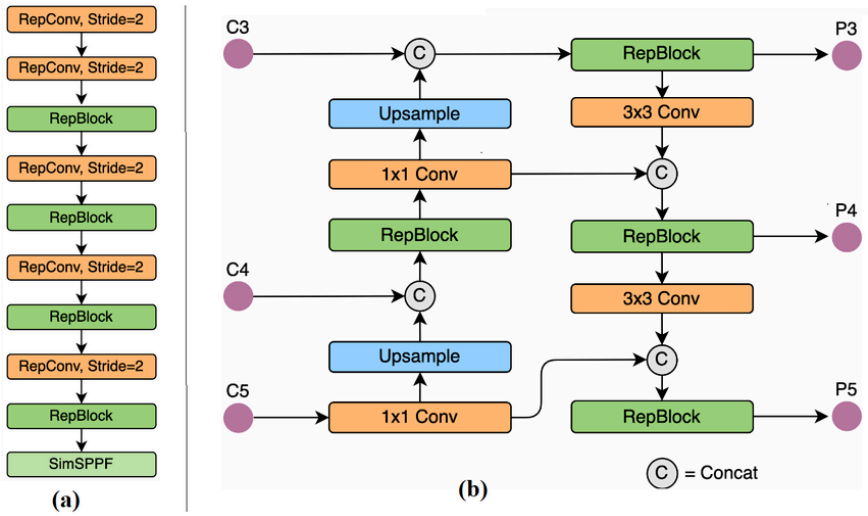
- **Triển khai dễ dàng**: YOLOv5 phát triển trên PyTorch, dễ tích hợp và triển khai trên nhiều nền tảng (ONNX, TensorRT).
- **Tốc độ cao**: Tối ưu tốc độ huấn luyện và suy luận, phù hợp cho các ứng dụng thời gian thực.
- **Data Augmentation**: Sử dụng Mosaic và MixUp giúp mô hình nhận diện tốt hơn trên dữ liệu đa dạng.
- **Neck với SPPF**: SPPF cải thiện khả năng nhận diện đối tượng có kích thước khác nhau mà không làm chậm tốc độ.
- **Dễ tùy chỉnh**: Hỗ trợ nhiều phiên bản mô hình từ nhỏ đến lớn, dễ dàng mở rộng.

2. YOLOv6

2.1. Kiến trúc:



2.2. Backbone: EfficientRep

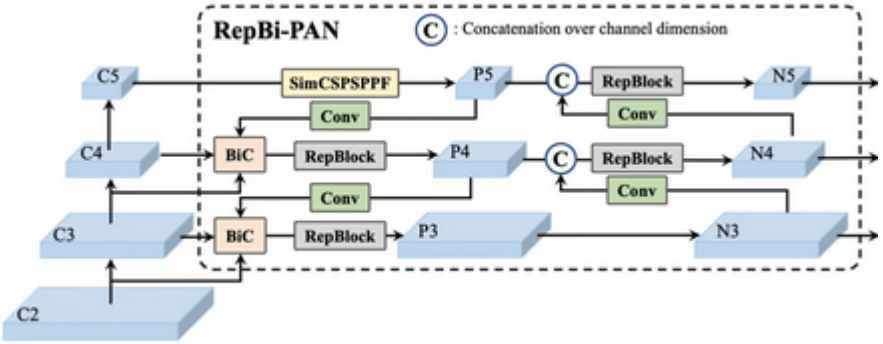


2.3. Activation function và Loss:

- Activation func: SiLU, ReLU
- Loss func:
  - Varifocal Loss (VFL): Được sử dụng cho bài toán phân loại.
  - Distribution Focal Loss (DFL): DFL được áp dụng cho bài toán hồi quy hộp

2.4. Neck:

- RepBi-PAN (Re-parameterized Bi-directional Path Aggregation Network)



(a) RepBi-PAN

- **C2, C3, C4, C5**: Đây là các đầu ra của các tầng backbone từ YOLO v6, đại diện cho các đặc trưng của ảnh ở các mức độ phân giải khác nhau (từ thấp đến cao). C2 có độ phân giải cao nhất, trong khi C5 có độ phân giải thấp nhất nhưng chứa nhiều thông tin trừu tượng.
- **BiC (Bidirectional Connection)**: BiC là kết nối hai chiều giữa các tầng đặc trưng. Nó giúp trao đổi thông tin qua lại giữa các tầng đặc trưng, giúp tối ưu hóa việc nắm bắt các đối tượng có kích thước khác nhau.
- **SimCSPSPPF**: Đây là một phiên bản đơn giản hóa của SPPF (Spatial Pyramid Pooling - Fast), được thiết kế để trích xuất đặc trưng từ các vùng không gian khác nhau một cách nhanh chóng và hiệu quả hơn.
- **RepBlock**: Đây là các khối re-parameterization, là các module tái cấu trúc giúp cải thiện hiệu suất của mạng mà không làm giảm chất lượng của việc phát hiện đối tượng. RepBlock có khả năng lưu trữ các phép biến đổi và tái cấu trúc của các khối CNN để giảm chi phí tính toán.
- **Concatenation over Channel Dimension (Ký hiệu C)**: Các đặc trưng sau khi được xử lý qua RepBlock và Conv sẽ được nối dọc theo chiều kênh (channel dimension), giúp tích hợp các đặc trưng từ nhiều tầng phân giải khác nhau.
- **N3, N4, N5**: Đây là đầu ra cuối cùng của các tầng đặc trưng từ neck, được tổng hợp từ nhiều mức phân giải khác nhau để cung cấp đầu vào tối ưu cho bước tiếp theo của YOLO v6.

2.5. Xử lý data

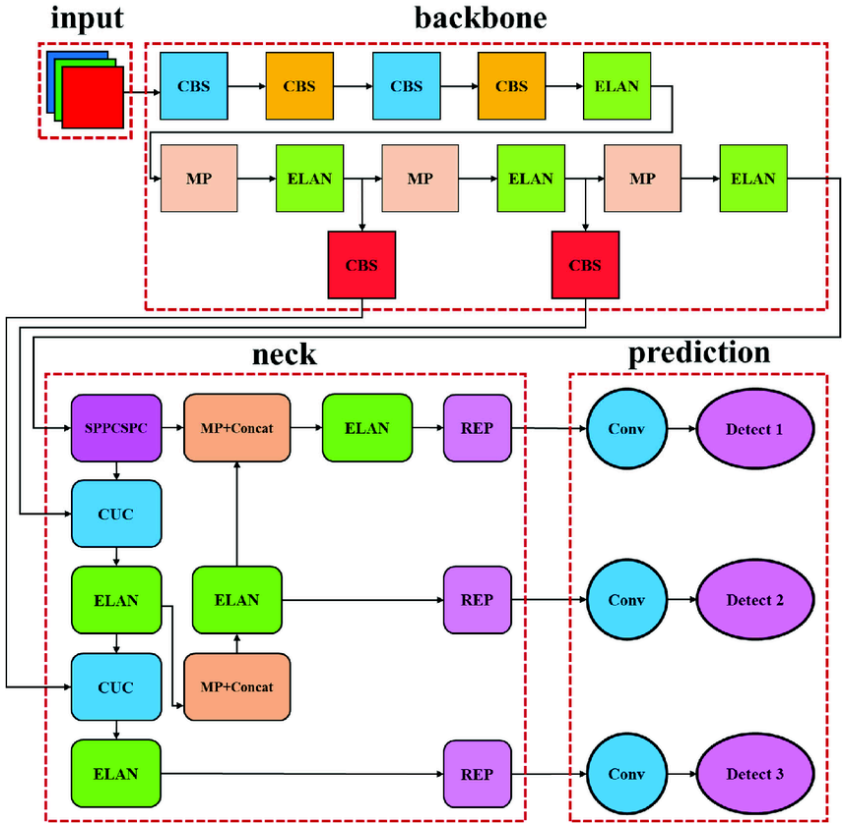
Không có gì đổi mới so với v5

2.6. Thay đổi so với bản trước

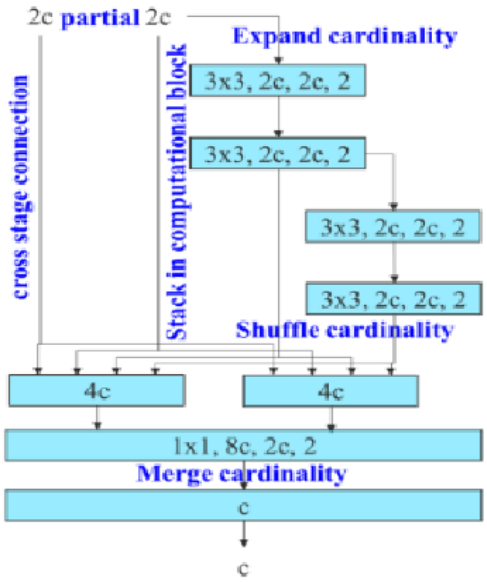
- **RepBi-PAN** trong phần neck, giúp cải thiện khả năng tổng hợp thông tin từ các feature map.

3. YOLOv7

3.1. Kiến trúc:



3.2. Backbone:



3.3. Activation function và Loss:

- Activation func: SiLU
- Loss func:
  - Objectness Loss: Đánh giá khả năng của mô hình trong việc phát hiện một đối tượng trong ảnh.
  - Classification Loss: Đánh giá độ chính xác trong việc phân loại các đối tượng được phát hiện.
  - Bounding Box Regression Loss: Đo lường độ chính xác của việc dự đoán vị trí hộp giới hạn (bounding box) cho các đối tượng.

3.4. Neck:

- SPPFCSPC (Spatial Pyramid Pooling-Fast, Cross-Stage Partial Channel)

3.5. Xử lý data:

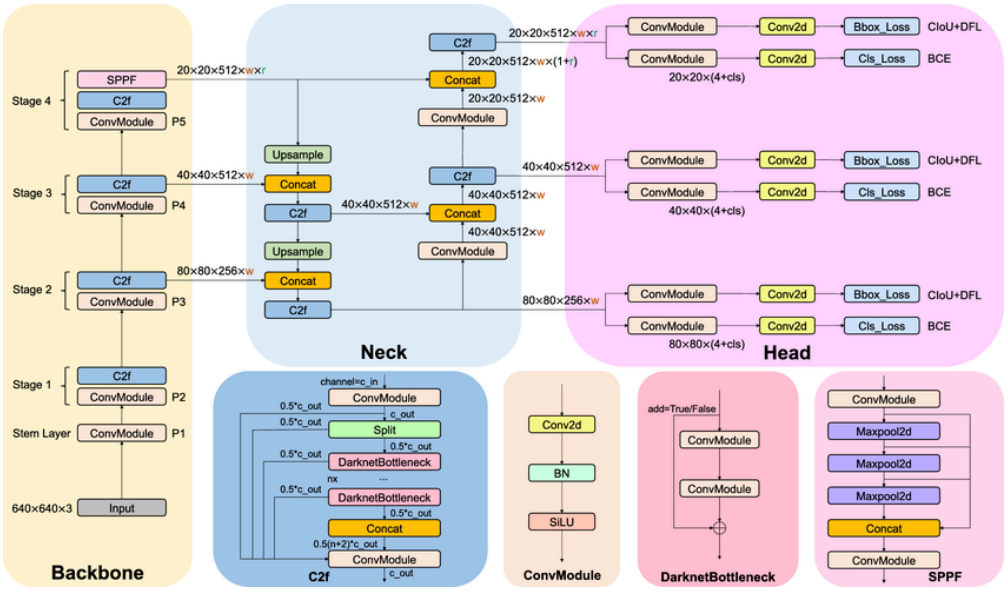
- Random Erasing: Kỹ thuật này ngẫu nhiên xóa một phần của hình ảnh, giúp mô hình trở nên kiên định hơn với các tình huống thiếu sót trong dữ liệu.
- Còn lại như các bản trước

3.6. Thay đổi so với bản trước

- E-ELAN cho phép tối ưu hóa cách mà các lớp tương tác, từ đó cải thiện khả năng học và giảm thiểu việc mất mát thông tin giữa các lớp.
- Cải thiện độ chính xác và hiệu suất.

4. YOLOv8

4.1. Kiến trúc:



4.2. Backbone:

- YOLOv8 áp dụng CSPDarknet, giúp cải thiện khả năng trích xuất đặc trưng và tối ưu hóa độ phức tạp tính toán.

4.3. Activation function và Loss

- Activation func: SiLU
- Loss func:

- Bounding Box Loss: Sử dụng IoU để đo lường độ chính xác giữa bounding box dự đoán và ground truth.
- Class Prediction Loss: Đo lường độ chính xác trong việc phân loại đối tượng, thường sử dụng hàm softmax.
- Objectness Loss: Đánh giá khả năng phát hiện đối tượng trong từng bounding box.
- Focal Loss: Được áp dụng cho cả objectness và class prediction để cải thiện khả năng phát hiện các đối tượng khó hoặc nhỏ.

4.4. Neck:

- Sự chuyển đổi sang PANet (Path Aggregation Network) cải thiện khả năng kết hợp đặc trưng giữa các lớp và nâng cao phát hiện đối tượng ở nhiều tỷ lệ.

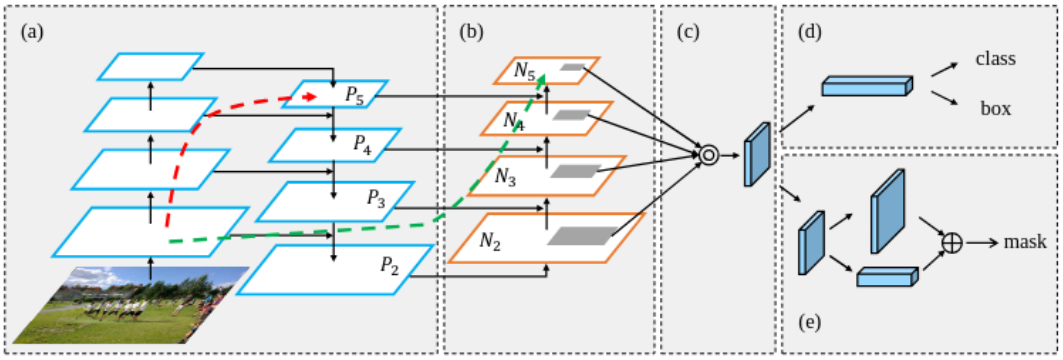


Figure 1. Illustration of our framework. (a) FPN backbone. (b) Bottom-up path augmentation. (c) Adaptive feature pooling. (d) Box branch. (e) Fully-connected fusion. Note that we omit channel dimension of feature maps in (a) and (b) for brevity.

4.5. Xử lý data

- MixUp: Kết hợp hình ảnh và nhãn để tạo ra các mẫu mới, giúp mô hình học được các đặc trưng phức tạp hơn.
- CutMix: Cắt một phần hình ảnh và chèn vào hình ảnh khác, giúp mô hình phát hiện tốt hơn các đối tượng có kích thước và hình dạng khác nhau.

4.6. Thay đổi so với bản trước

- Độ Chính Xác Cao: Cải tiến trong kiến trúc giúp phát hiện đối tượng chính xác hơn, ngay cả trong điều kiện khó khăn
- Tốc Độ Nhanh: Tối ưu hóa cho suy diễn thời gian thực, phù hợp cho giám sát và phản ứng khẩn cấp
- Khả Năng Tổng Quát Tốt: Sử dụng các kỹ thuật augmentation như MixUp và CutMix giúp mô hình xử lý tốt hơn các trường hợp chưa thấy

- Phát Hiện Đối Tượng Nhỏ: Cải thiện khả năng phát hiện các đối tượng nhỏ và khó, mở rộng ứng dụng