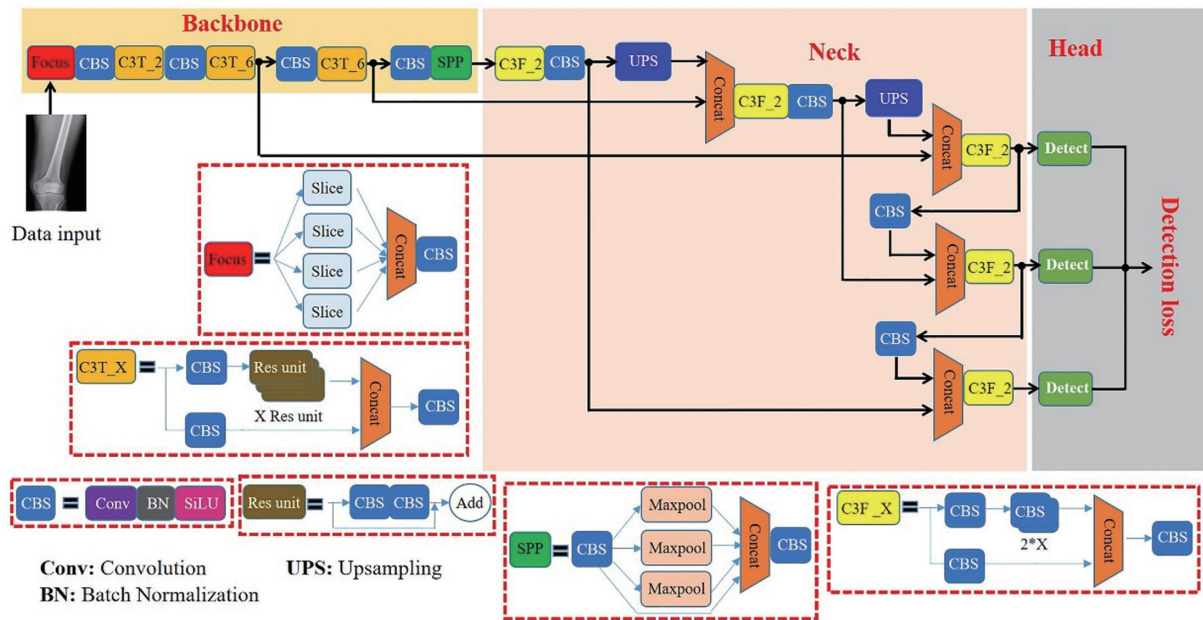## Architecture of YOLOv5

Training parameters

The training batch -size was 16. For the SGD optimiser, we used Complete IoU Loss, BCEWithLogits Loss and Focal Loss to calculate bounding box position loss, classification loss, and confidence loss, respectively. The initial learning rate was 0.01. In the warm-up stage, the learning rate adjustment strategy of one-dimensional linear interpolation was adopted; after this stage, the cosine annealing was applied. The IoU threshold of the final non-maximum suppression was set to 0.6. The model was trained for 300 epochs.

Model training was performed with Python 3.8.5 (https://www.python.org/) and PyTorch 1.7.1 (https://pytorch.org/) using an NVIDIA TITAN RTX 24GB graphics processing unit. PyTorch tightly integrated with NVIDIA's CUDA for fast computation on GPUs. PyTorch integrated with other scientific computing libraries in Python, such as NumPy, providing convenience for data preprocessing and analysis. PyTorch provided support for multi GPU training, which is crucial for our model training.

Details of test-time augmentation method

During testing, each image was left-right flipped and scaled to 1062 * 1062 and 858 * 858 (scaled images were padded to the same size of 1280 * 1280). The DL model predicted the two augmented images and the original test samples. The three outputs were merged before non-maximum suppression.