TVM BENCHMARK

2025-10-21 김규진

TVM BenchMark (No Parameter Load)

Conv

Relu forward

```
================================================================================
OVERHEAD BREAKDOWN
================================================================================

[Test 1] Execution with pre-unpacked parameters
  Time: 0.263273 ms

[Test 2] Parameter unpacking overhead only
  Time: 0.000149 ms

[Test 3] Impact of batch size
  Batch   1: TVM=0.2582ms, PyTorch=0.0158ms, Speedup=0.06x
  Batch   8: TVM=1.7490ms, PyTorch=0.0334ms, Speedup=0.02x
  Batch  32: TVM=6.9064ms, PyTorch=0.0354ms, Speedup=0.01x
  Batch  64: TVM=13.7818ms, PyTorch=0.0641ms, Speedup=0.00x
```

Resnet 18

```
============================================================
TVM CPU-OPTIMIZED INFERENCE TIME (over 100 runs)
============================================================
  Average: 1959.64 ms
  Std Dev: 22.03 ms
  Min:     1914.91 ms
  Max:     2058.70 ms
============================================================

✓ Inference completed - Output shape: (1, 1000)

============================================================
TOP-5 PREDICTIONS FOR DOG IMAGE
============================================================
1. Pomeranian                39.73%
2. Samoyed                   22.64%
3. keeshond                   6.05%
4. Japanese spaniel           5.42%
5. Arctic fox                 4.58%
```

```
============================================================
PYTORCH INFERENCE TIME (over 100 runs)
============================================================
  Average: 13.89 ms
  Std Dev: 1.69 ms
  Min:     11.18 ms
  Max:     19.58 ms
============================================================

✓ Inference completed - Output shape: (1, 1000)

============================================================
TOP-5 PREDICTIONS FOR DOG IMAGE
============================================================
1. Pomeranian                39.73%
2. Samoyed                   22.64%
3. keeshond                   6.05%
4. Japanese spaniel           5.42%
5. Arctic fox                 4.58%
```

TVM vs Pytorch

- TVM (element-wise) vs Pytorch (conv)
- TVM 은 단일 연산에 이점이 있음 (fusion)
- Pytorch 는 병렬연산에 이점이 있음 (im2col + GEMM , cublas)
- 일반적인 모델은 Conv 의 연속인데 TVM 이 유리할까?

TVM BenchMark

- Convolution (NO Auto Tuning)

```
============================================================
Summary
============================================================
TVM Relax average time: 1.321700 ms
PyTorch average time:   0.920520 ms

PyTorch is 1.44x faster than TVM
============================================================
```

Convolution (Auto Tuning)

```
  ================================================================
  Summary
  ================================================================
  TVM (Auto-Tuned)          0.259106 ms
  PyTorch                   0.911648 ms

  ✓ TVM is 3.52x FASTER than PyTorch
  ================================================================
(tvm-build-venv) gyujinkim@Gyujinui-Macmini jr %
```

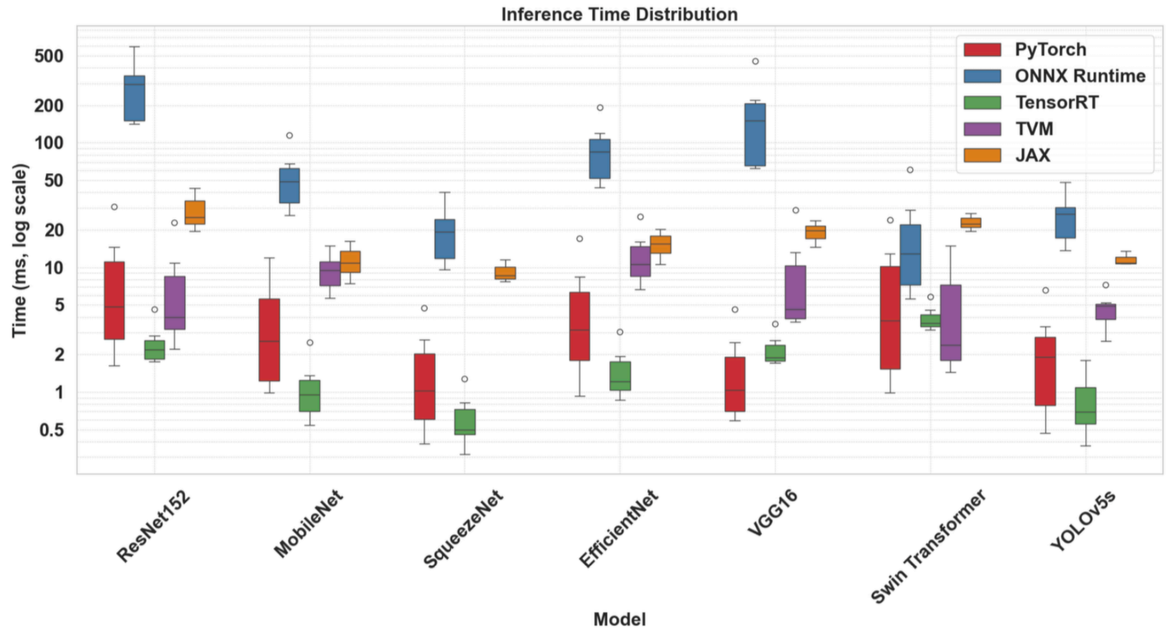
진행사항

- PyTorch: Input → C++ Kernel → Output
- TVM: Input → VM Interpreter → Compiled Function → Output
- MetaSchedule (AutoTuning)? -> 이게 핵심인데 아키텍쳐에 맞게 operator 들을 fusion해서 연산별로 딜레이체크해서 최적의 속도 op 를 찾음 (tiny engine과 유사) 현재는 tvm relax 의 스케쥴링이 모델 로그 스케일이 커짐에 따라 런타임 딜레이가 줄어들도록 설계하고있음 but 아직 0.22dev 버전이기때문에 지원하지 않는 op가 많아 큰 모델을 테스트해볼 수가 없음

참고

- https://www.mdpi.com/2079-9292/14/15/2977?utm_source=chatgpt.com

Figure 1. Inference Time Distribution (y axis in log scale).



다음 단계

- Tensor RT Pipeline
- TVM Grahph + Auto Scheduling