



CCF BDCI

# 2024 CCF 大数据与计算智能大赛<sup>12th</sup>

---

## 决赛答辩评审会

《基于TPU平台的OCR模型性能优化》

队伍名称：识唔识得

# 目录



CCF BDCI

## 01 赛题分析

## 02 解决方案

开源调研

模型迁移

方案评估

## 03 实验结果

## 赛题分析

在**低端嵌入式设备**上部署**文本识别模型**（复杂街景场景）

利用 TPU 平台实现 OCR 模型量化部署，落地端侧场景

- 基线项目: [ppocrv3@fp16 CV186X](#)
  - CPU: 8 core ARM, TPU: 16 TOPS INT8
- Milk-V Duo 系列开发板: Duo 64MB (CV1800B)
  - CPU: 1 core RISC-V, TPU: 0.5 TOPS INT8

评分公式: 文字识别质量 (精度), 模型部署成本 (推理速度)

- $score = 90 + 40 * f1\_score - 0.085 * infer\_time$
- 满分线规格
  - f1\_score: 0.57 (ppocrv3 官方参考数据)
  - infer\_time: 150ms

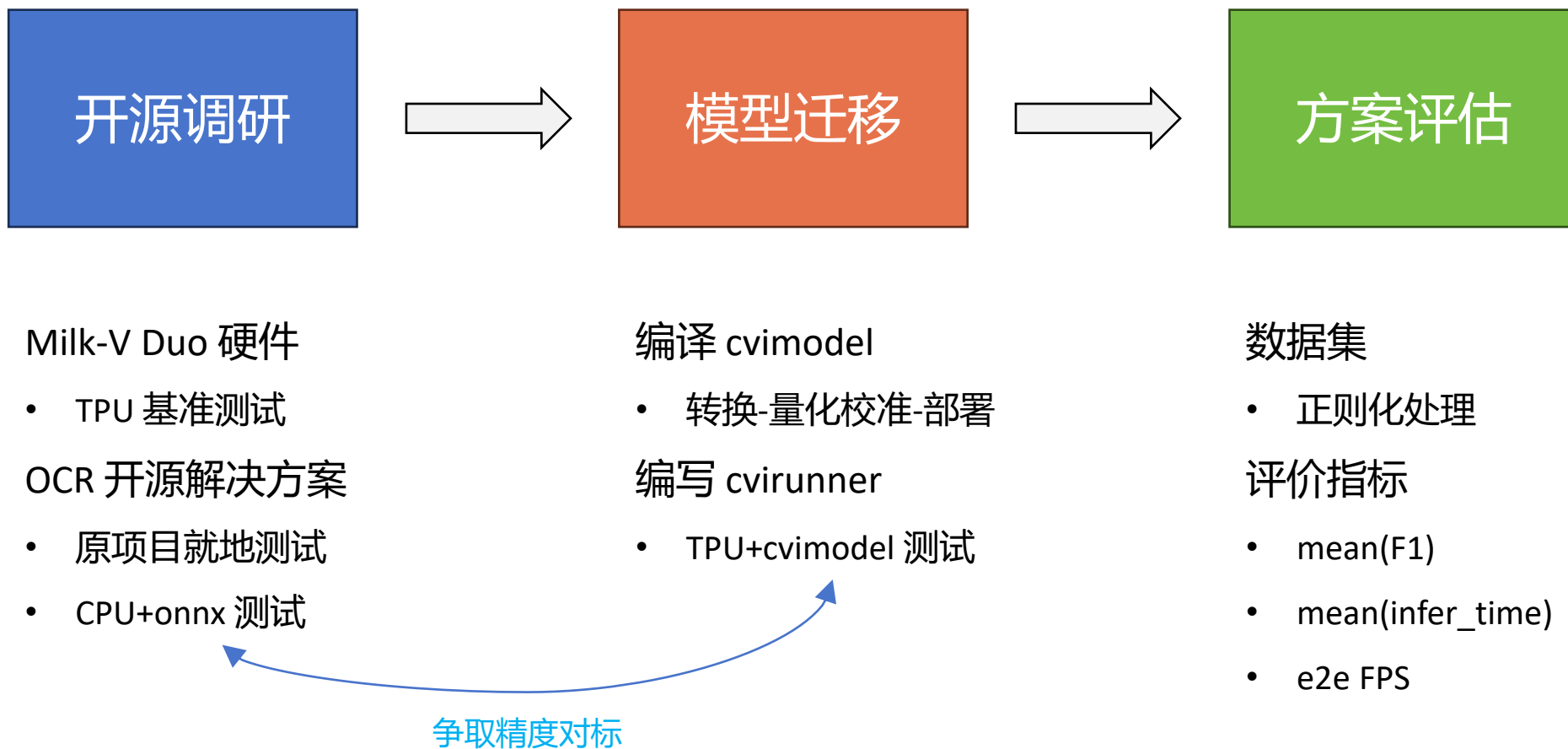


家管修镇  
堂美容美发  
伍家山茶



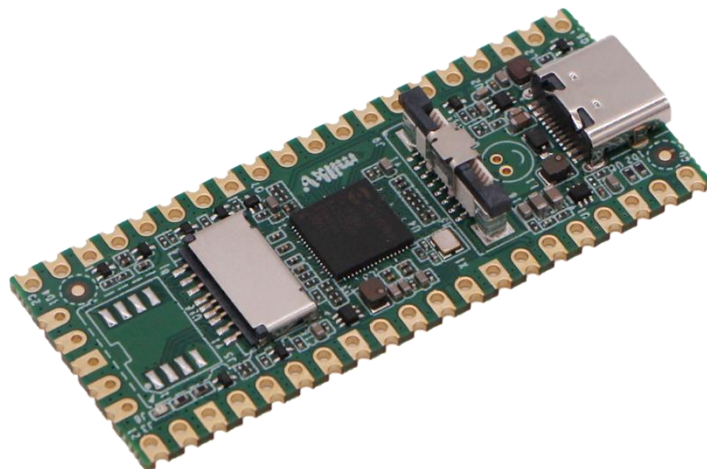
中国 雄梅石材  
极光宽带报装点  
广州市白云区同德同雅东街C9铺  
服务热线: 13824415526  
故障报修电话: 15718382777  
联系电话: 13924301893 13660388477 梁生

## 解决思路



## 硬件调研

- [Milk-V Duo \(CV1800B\)](#)
  - CPU: RISC-V C906@1.0 GHz
  - TPU: 0.5 TOPS@INT8
  - Mem: DDR2 16bit 64M (26.8M reserved for ION)



MilkV Duo 开发板

- TPU benchmark

模型	GFLOPS	运行时间 (s)	[FPS]
resnet18	1.81	0.30	3.33
densenet121	2.83	0.85	1.18
mobilenetv2	0.30	0.20	5.00
shufflenetv2	0.14	0.16	6.25
squeezenet1.1	0.35	0.16	6.25
googlenet12	1.5	0.24	4.16
yolov5n	4.5	0.55	1.80

(\*) 运行时间用time工具测定：含图像与模型的加载时间

(\*\*) GFLOPS 数据来自 torchvision 参考

<https://pytorch.org/vision/stable/models.html>



## 模型调研

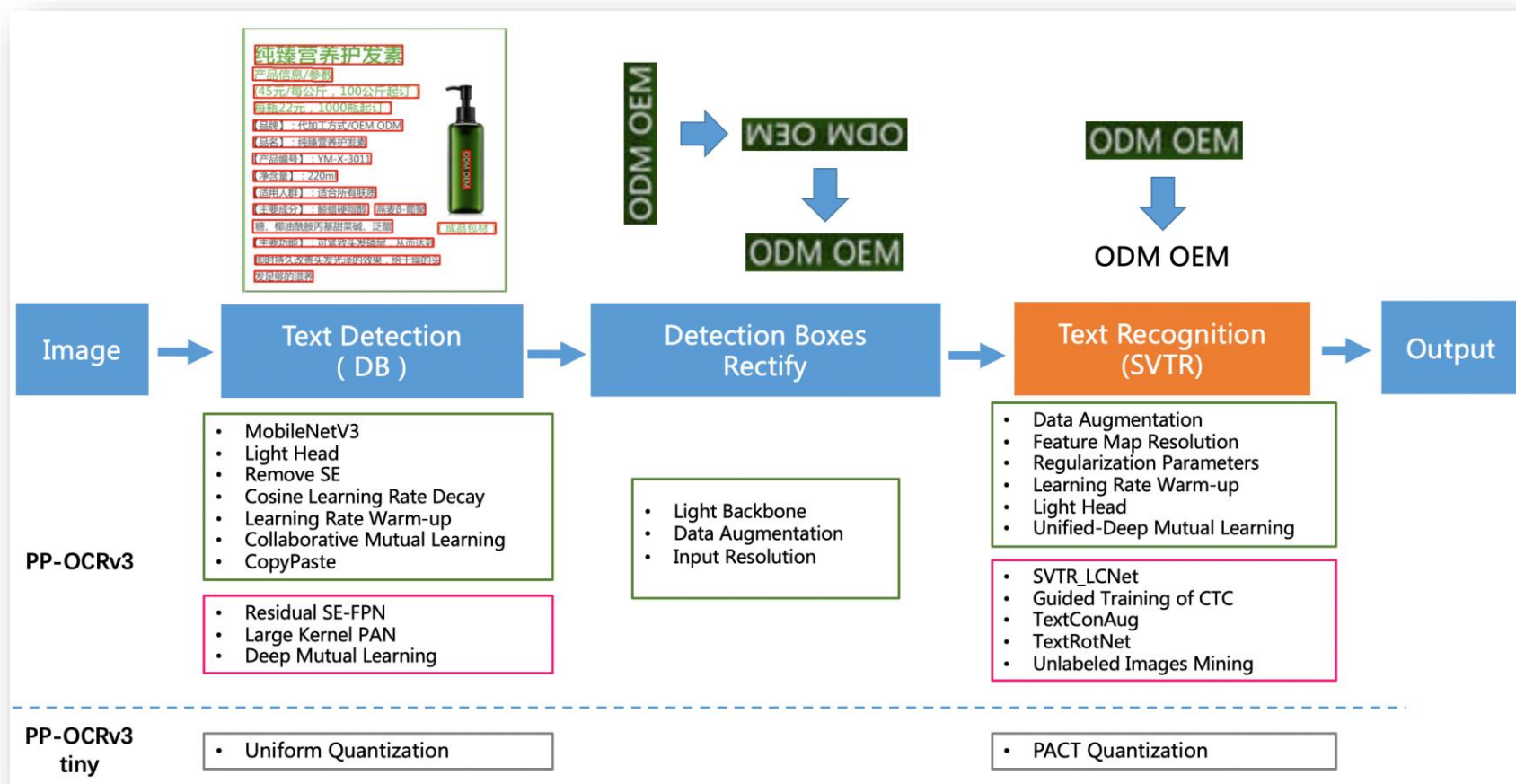
- 轻量化开源 OCR 解决方案

- [PaddleOCR](#)
- RapidOCR
- CnOCR
- chineseocr\_lite

- PPOCR 流水线

- det 文本框检出
- cls 文本框旋正
- rec 文本框识别

$$\text{ocr} := \text{det} * 1 + (\text{cls} + \text{rec}) * k$$



PPOCR v3 架构图

[https://paddlepaddle.github.io/PaddleOCR/latest/ppocr/blog/PP-OCRv3\\_introduction.html](https://paddlepaddle.github.io/PaddleOCR/latest/ppocr/blog/PP-OCRv3_introduction.html)

模型调研

- PPOCR benchmark (CPU+onnx)

发行板本	det	rec	cls	官方参考 Hmean	A榜 推理时间 (ms)	A榜 F1-Score
v4	DBNet (4.51MB)	SVTR (10.3MB)	mbnetV3 (559KB)	62.24%	76.05	0.60724
v3	DBNet (2.31MB)	SVTR (10.1MB)		57.99% 62.90%	58.68	0.57585
v2	DBNet (2.22MB)	CRNN (7.99MB)		57.60%	43.02	0.52051
mobile	DBNet (2.22MB)	CRNN (4.22MB)			41.61	0.34883

- 迁移与部署的出发点
  - 可暂时忽略 cls 模型：作用不是很大，且导致较高时延
  - 优先考虑 v2 系列：v3/v4 模型较大，转换后大概率炸 ION 内存

## 模型转换与编译: paddle $\rightarrow$ onnx $\rightarrow$ mlir $\rightarrow$ cvimodel

- ONNX 优化
  - 移除模型末尾的 Softmax 层
- 转换 (transform)
  - 设定 mean/scale/input\_shapes
- 校准 (calibrate)
  - 使用 300 个样例
  - 对 rec 校准数据集做右端填充 255
- 部署 (deploy)
  - 量化数据类型
    - det: INT8 sym + quant\_output
    - rec: **BF16** (混精度收益不大)
  - 前处理融合
    - 输入格式: BGR\_PACKED (符合 opencv::imread)

```
model_transform.py \  
  --model_name $MODEL_NAME \  
  --model_def $MODEL_DEF \  
  --input_shapes $INPUT_SHAPE \  
  --mean $MEAN \  
  --scale $SCALE \  
  --keep_aspect_ratio \  
  --test_input $TEST_INPUT \  
  --test_result $TEST_RESULT \  
  --debug \  
  --mlir $MLIR_MODEL_FILE
```

```
run_calibration.py $MLIR_MODEL_FILE \  
  --dataset $CALI_DATASET \  
  --input_num 300 \  
  -o $CALI_TABLE_FILE
```

```
model_deploy.py \  
  --chip $CHIP \  
  --mlir $MLIR_MODEL_FILE \  
  --quantize $QTYPE \  
  --quant_input \  
  $QUANT_OUTPUT \  
  $QUANTIZE_TABLE \  
  --calibration_table $CALI_TABLE_FILE \  
  --test_input $TEST_INPUT \  
  --test_reference $TEST_RESULT \  
  --tolerance 0.85,0.45 \  
  $COMPARE_ALL \  
  --fuse_preprocess \  
  --customization_format BGR_PACKED \  
  --ignore_f16_overflow \  
  --op_divide \  
  --debug \  
  --model $CVI_MODEL_FILE
```



# 模型转换与编译: paddle → onnx → mlir → cvimodel

- 编译产生 cvimodel 的基本信息

子模型	编译预设	GFLOPS (mlir)	ION 内存需求 (MB)	板上推理时间 (ms)
det	v4_det_int8	4.904	22.26	OOM
	v3_det_int8	3.918	14.10	270.805
	v2_det_int8	3.906	13.25	247.386
	mb_det_int8	3.906	13.25	223.39
	v2_det_int8_480	-	6.90	128.963
	v2_det_int8_320	-	3.72	46.317
rec	v3_rec_bf16	0.942	10.07	95.523
	v2_rec_bf16	1.130	12.42	65.314
	mb_rec_bf16	0.286	7.93	33.612
cls	mb_cls_int8	0.119	0.77	-



优先  
考虑



## 模型运行时 cvirunner

- 基于 cviruntime ([milkv-duo/tpu-sdk-cv180x](https://github.com/milkv-duo/tpu-sdk-cv180x)) 实现
- det 前处理: resize + pad=0
- det 后处理: binarize + findContours + **unclip**
- rec 前处理: resize + pad=255
- rec 后处理: argmax
- **黄色**标记的函数调用将用作**性能计时点**
  - ts\_model\_load / ts\_model\_unload
  - ts\_img\_load / ts\_img\_crop
  - ts\_det\_pre / ts\_det\_infer / ts\_det\_post
  - ts\_rec\_pre / ts\_rec\_infer / ts\_rec\_post

```
Model det = load_model(), rec = load_model();

for (string &file : file_list) {
    Mat img = imread(file);

    Mat img = det.preprocess(img);
    int8 *segmap = det.infer(img);
    vector<Box> box_list = det.postprocess(segmap);

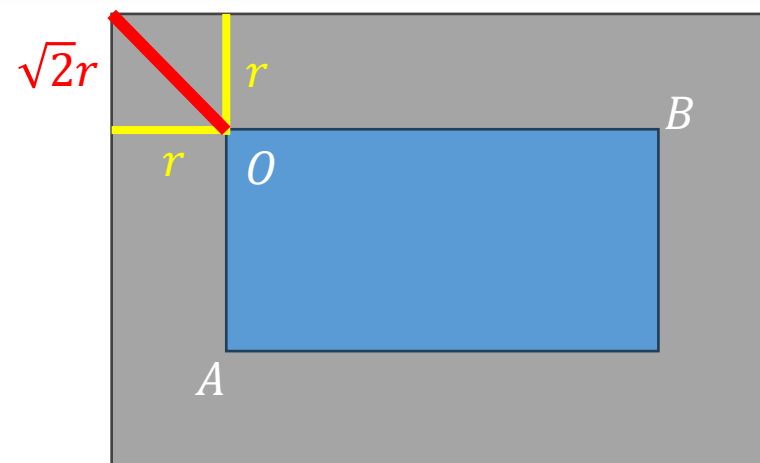
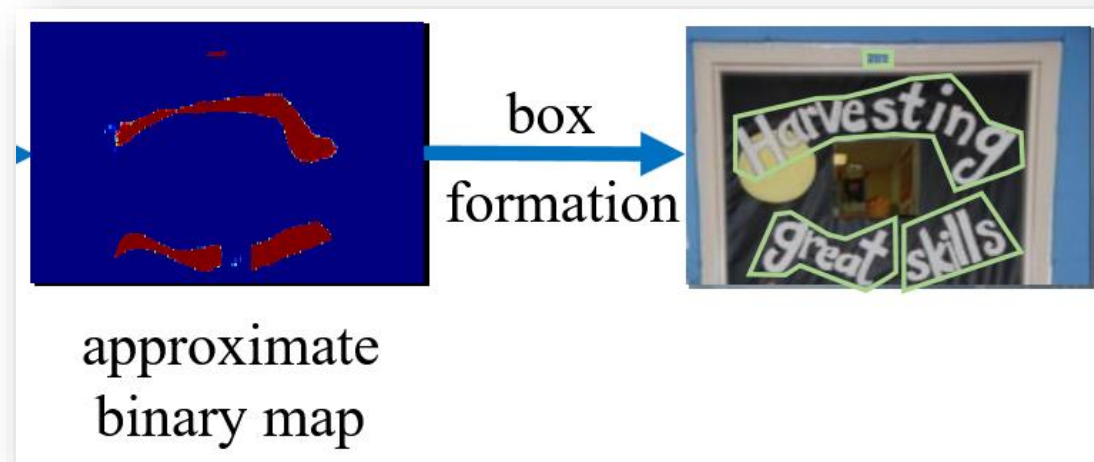
    for (Box &box : box_list) {
        Mat img_crop = warp_crop_perspective(img, box);

        Mat img_crop = rec.preprocess(img_crop);
        float32 *logits = rec.infer(img_crop);
        int[] token_ids = rec.postprocess(logits);
    }
}

unload_model(det); unload_model(rec);
```

## 模型运行时 cvrunner – unclip 算法的近似实现

- 考虑增加外边距  $r$ , 使矩形框面积扩大  $k$  倍
  - 即有  $(w + r)(h + r) = kwh$
  - 解得  $r = \frac{1}{4} \left[ \sqrt{(w + h)^2 + 4(k - 1)wh} - (w + h) \right]$
- 各顶点按法向外移  $\sqrt{2}r$  距离
  - 设顶点  $O$  的左右邻居为点  $A$ 、 $B$
  - 定义法向  $\vec{v} = \text{norm}(\overrightarrow{AO} + \overrightarrow{BO})$
- 减少了第三方库依赖 ☺
  - $k$  为超参数 (默认值 2.7)



unclip 示意

## 实验设置

- 数据集
  - A榜: ICDAR2019-LVST, 2350 个样本
  - B1榜: MSRA-TD500, 500 个样本
  - B2榜: 未知数据集, 3992 个样本; 对于尺寸超过 640 的图预先降采样处理
    - 我们的Duo板子不支持 > 1.5GB的磁盘分区, 无法上传原数据集 (3.91GB)
    - JPEG解码内存不足导致严重的 MemSwap
      - 磨损 TF 寿命; 干扰计时测定; 概率炸内存 SegFault
- 评估指标: 比赛给定样例项目中的评定规则
  - mean(F1): 以文本框为单位,  $\text{box\_iou} > 0.5$  且  $\text{text\_sim} > 0.5$  记为一个 TP
  - mean(infer\_time): 平均一次det + 一次rec的纯TPU推理时间
  - real\_fps: 不计数据读取时间的端到端 FPS

```
ts_model_load: 829.000 ms
ts_model_unload: 108.380 ms
=====
n_img:          3992
n_crop:         11920
-----[Total]-----
ts_img_load:    384295.531 ms
ts_img_crop:    94681.820 ms
ts_det_pre:     20239.100 ms
ts_det_infer:   885796.500 ms
ts_det_post:    93333.594 ms
ts_rec_pre:     19606.209 ms
ts_rec_infer:   398918.812 ms
ts_rec_post:    77933.859 ms
-----[Average]-----
ts_det_pre:     5.070 ms
ts_rec_pre:     1.645 ms
ts_pre:         9.981 ms
ts_det_infer:   221.893 ms
ts_rec_infer:   33.466 ms
ts_infer:       321.822 ms
ts_det_post:    23.380 ms
ts_rec_post:    6.538 ms
ts_post:        42.903 ms
=====
Total time:     1977897.125 ms
```

cvirunner给出的原始计时统计

评估结果

我们的提交材料中包含更多的可用模型组合，可按需任意测试 ☺


数据集	模型配置	F1	infer_time	e2e FPS	estimate/submit score
A榜	v2_det + v2_rec	0.44099	333.937	0.88	79.25498
	v3_det + mb_rec	0.42010	277.942	1.22	83.17896
	v2_det + mb_rec	0.42781	256.211	1.42	85.33433
	v2_det + mb_rec (480)	0.33901	155.279	1.885	90.36170
	v2_det + mb_rec (320)	0.20613	75.951	2.954	91.78934
	mb_det + mb_rec	0.32475	256.930	1.47	81.15095

主观最优

最佳平衡



## 提交得分

A 榜		B 榜						
排名	排名变化	队伍名称	有效提交次数	最高分提交时间	最高得分	f_core	max f_core	
 	-	识唔识得	1	2024-11-16 18:44	99.38789602	0.44719740	0.44719740	
	-	常务副SOTA	1	2024-11-16 22:46	99.36871660	0.44671792	0.44671792	
	↓ 1	default7629265	2	2024-11-16 23:48	98.25850707	0.41896268	0.41896268	
4	-	1024	2	2024-11-15 08:19	89.94687500	0.00000000	0.00000000	
5	-	tcco	1	2024-11-16 23:40	87.25703265	0.14392582	0.14392582	

注：评测网站无法提交infer\_time，其反馈的分数并非按定义的评分公式计算 🤔

推理样例 (印刷字体/密集, 良好 🧐)



家乐福中关村广场

免费送货车

大宗购物热线: 51721515

CARREFOUR ZGC PLAZA STORE

FREE-DELIVERY

BIG PURCHASE SERVICE LINE 51721515

## 推理样例 (印刷字体/密集/竖排文本, 次之 😊)



国家自主创新示范区核心区  
中关村商务中心区品牌发布专业平台

招商  
热线  
“  
线

010-82483330

13391699998

13911615850

优京人系世纪园除店告育限公司

DIVINESKY

尽精微·致广大

www.86ad.com



## 推理样例 (长文本/小字/反光, 不太好 🤔)



及用存较在水中食，网将定则在暂冰箱卫生，并理变质食

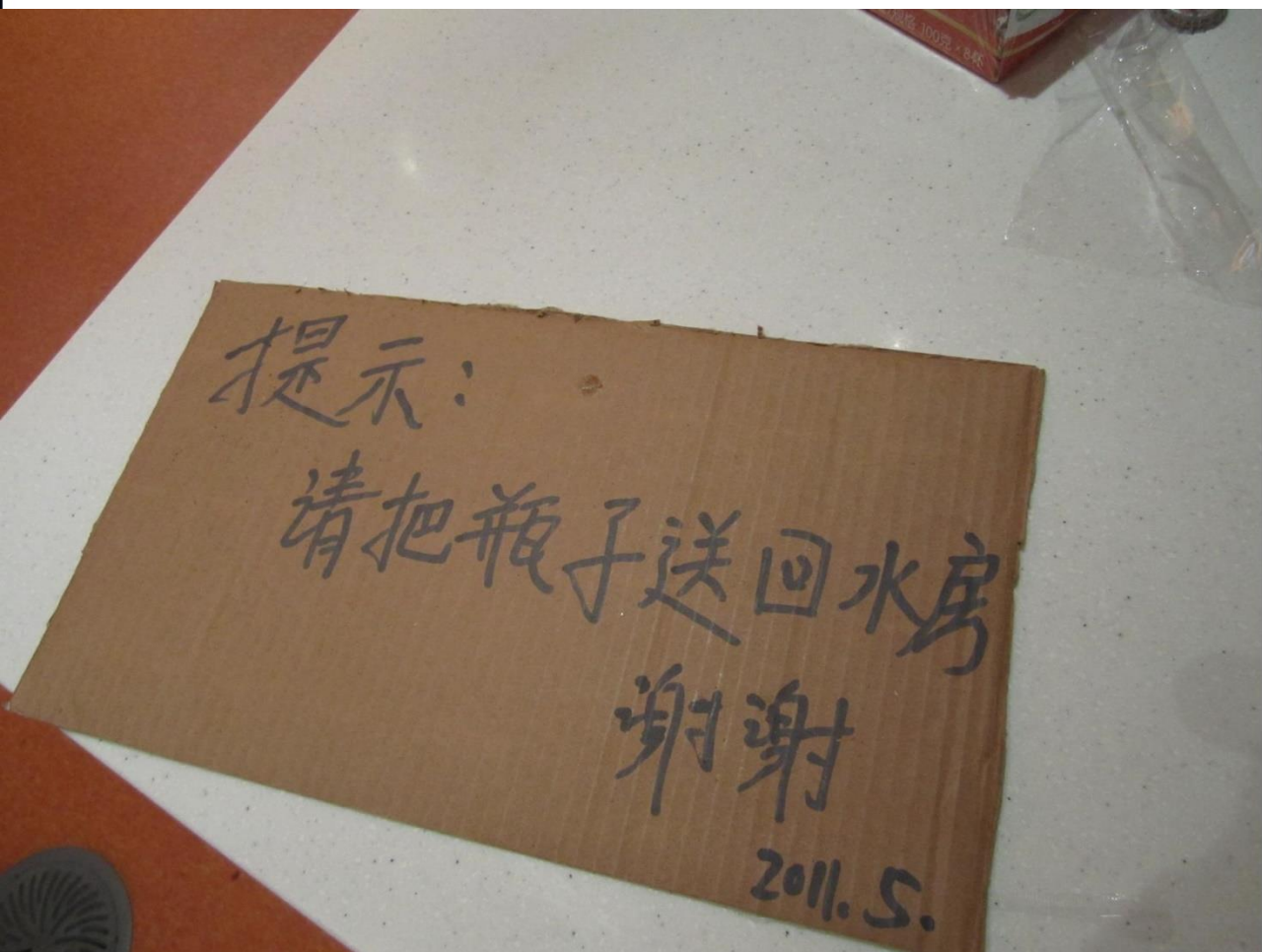
rdieoiwildsgoseheroen

微款房地产及设施管理

MS RE&FTea

veing

## 推理样例 (手写字体, 不太好 🤪)



很示:

请他液于送回水房

湘梨

ZLS



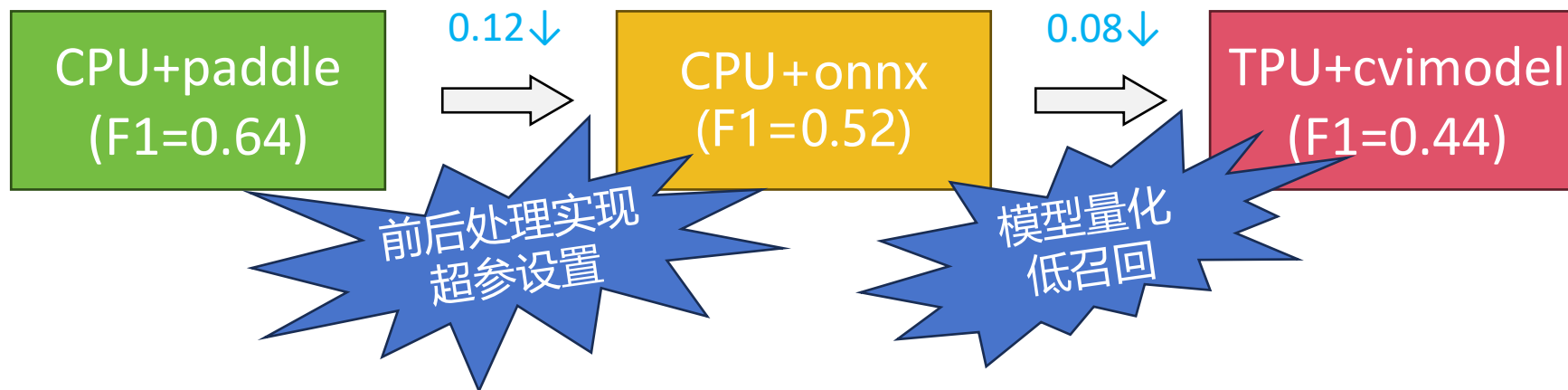
推理样例 (艺术字体/透视畸变, 离谱 🤪)



一奶杀  
0itlreao

## 讨论 & 总结

- 讨论：迁移后精度损失来源
  - 以 ppocrv2+A榜数据 为基准



- 总结
  - 将 ppocr 系列模型移植到 Milk-V Duo (CV1800B) 板上部署运行, 支持任意组件混搭
  - 优化 cvimodel 编译参数, 优化数据校准方式
  - 实现高效的 cvirunner, 提出近似 unclip 算法
  - 3种运行环境下的模型基准测试, 分析模型迁移前后的精度损失来源



CCF BDCI

# 2024 CCF 大数据与计算智能大赛<sup>12th</sup>

---

感谢观看!