Supplementary Material for: Portmanteauing Features for Scene Text Recognition

Yew Lee Tan, Ernest Yu Kai Chew and Adams Wai-Kin Kong
School of Computer Science and Engineering
Nanyang Technological University
Singapore

Jung-Jae Kim and Joo Hwee Lim
Institute for Infocomm Research
Agency for Science, Technology and Research
Singapore

I. STN

A. Input Protocols for Portmanteau Features

To generate portmanteau features, the raw images were first converted into grayscale images, before being normalized via two transformation protocols.

The first protocol resized the image to a fixed height of 32 pixels with varying width to retain the aspect ratio. After this resizing, if the width of the image was shorter than 128 pixels, zero padding would be applied. Otherwise, the width of the image would be further resized to 128 pixels.

The second protocol directly resized the input image to 300×300 pixels for STN and 32×128 pixels for MORN. The size of the output images from the two rectification networks is 32×128 pixels. Both protocols would yield grayscale images with a size of 32×128 pixels.

The STN was trained with a batch size of 32 for 150 epochs, with a configuration of initial learning rate as $5e^{-4}$, the warm-up iteration was $10,000$, and the decay per iteration was set such that the final learning rate is one-fifth of its initial. The Adam optimizer used was configured with betas of $(0.9, 0.999)$, and an eps of $1e^{-8}$. Lastly, the STN dataset contains $80,000$ synthetic images with a 9-to-1 train-validation split.

In addition, the STN also adopts a custom loss function, which is defined as:

$$L_{STN} = || \psi - \hat{\psi} || + \alpha || \phi - \hat{\phi} || \\ + \beta || \cos\theta - \cos\hat{\theta} || + \gamma || \sin\theta - \sin\hat{\theta} || + \delta || \xi - \hat{\xi} || \tag{1}$$

where $|| \bullet ||$ represents L1 norm, $\hat{\psi}$ is the estimated coefficients of the 4th order Legendre polynomial, $\hat{\theta}$ is the estimated angles between the line segments and the tangents of the polynomial, $\hat{\phi}$ is the estimated x-coordinates of the intersection points between the line segments and the polynomial, and $\hat{\xi}$ is the length of the line segments. ψ, θ, ϕ, and ξ are the corresponding ground truths and α, β, γ, and δ are hyperparameters. It should be highlighted that the area between the predicted and ground-truth polynomial is directly proportional to the difference between their Legendre coefficients.

B. Polynomial Scheme for STN

The polynomial scheme could be dividing into six parts, as follows:

1) extracting the key points from the ground-truth bounding boxes
2) fitting the polynomial curve and lines, using the extracted key points
3) determining the intersection points between the polynomial and its line segments
4) interpolating the intersection points and the line angles to get the desired number of line segments
5) converting the coefficients of the fitted polynomial into its Legendre equivalence
6) combining all the variables into a single loss function

Key points extraction: The first step in the polynomial scheme was to resize the image into a square image. The height and width of the square image were considered to be in the range of -1 and 1.

Subsequently, the top-center, center and bottom-center of each bounding box, in addition to the top, center and bottom of the left-most and right-most corners, were extracted as key points for curve and line fitting. These key points were denoted as the green and red points in Figure 1b.

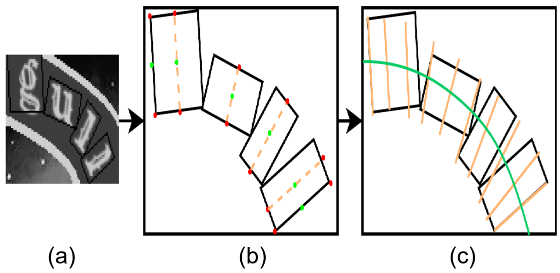

Fig. 1. Overview of the STN polynomial scheme, where the ground-truth bounding boxes were used to derive a 4th order polynomial with 10 lines segments.

Curve and line fitting: After which, the center key points were used to fit a 4th order polynomial, while the top, center, and bottom points were used to fit $(n+2)$ linear functions, where n was the number of characters in the word.

Intersection between polynomial and line: Then, the intersection points between the polynomial and the $n+2$ line segments were determined and denoted as ϕ_{raw}. The angles between the tangent lines and the line segments were also calculated and denoted as θ_{raw}. In addition, the height of the tallest bounding box was computed and denoted as ξ.

Line segments interpolation: Next, ϕ_{raw} and θ_{raw} would be used as key reference points for rotational and translational interpolation to produce the desired number of line segments (from $n+2$) of length ξ. In this paper, the desired number of line segments was set to 10 (See Figure 1c).

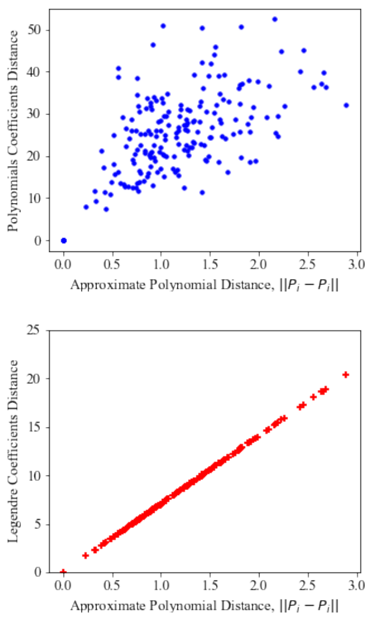

Fig. 2. The scatter plots of polynomial coefficients distance (top) and Legendre coefficients distance (bottom) of 400 polynomial pairs against $\|P_i - P_j\|$.

Legendre polynomial: Then, the coefficients of the polynomial would be converted to its Legendre equivalence. This is because Legendre polynomials form a complete orthogonal basis on L^2 [-1,1] (L^2 space) [1] and so, their coefficients may be better for measuring the similarity between two polynomials.

Figure 2 was constructed with 20 random polynomials $f_i(x)$, where $i \in [1, 20]$. Each polynomial was paired with each other to form 400 polynomial pairs. $N = 201$ evenly distributed points across the x-axis from $[-1, 1]$ were sampled from each polynomial and the y values form a point, P_i, where $P_i \in \mathbb{R}^N$. The 400 polynomial pairs were represented as in the scatter plots, in terms of the distance, $\|P_i - P_j\|$, and the $\|coef(f_i) - coef(f_j)\|$, where $i, j \in [1, 20]$ and $coef()$ returns the coefficients of its polynomial function.

Although the polynomial coefficient distances seemly correlate with the distance $\|P_i - P_j\|$ (top of Figure 2), they did not demonstrate proportionality like its Legendre equivalence (bottom of Figure 2), which was the reason Legendre coefficients were used in this work instead.

Loss function:

$$L_{STN} = || \psi - \hat{\psi} || + \alpha || \phi - \hat{\phi} ||$$
$$+ \beta || \cos\theta - \cos\hat{\theta} || + \gamma || \sin\theta - \sin\hat{\theta} || + \delta || \xi - \hat{\xi} || \quad (2)$$

where $|| \bullet ||$ represents L1 norm, $\hat{\psi}$ is the estimated coefficients of the 4th order Legendre polynomial, $\hat{\theta}$ is the estimated angles between the line segments and the tangents of the polynomial, $\hat{\phi}$ is the estimated x-coordinates of the intersection points between the line segments and the polynomial, and $\hat{\xi}$ is the length of the line segments. ψ, θ, ϕ, and ξ are the corresponding ground truths and α, β, γ, and δ are hyperparameters.

C. STN configurations

Layers	Output Size	Configurations
ConvBlock1	$32 \times 50 \times 50$	$3\times3, BN, ReLU, 2\times2$
ConvBlock2	$64 \times 25 \times 25$	$3\times3, BN, ReLU, 2\times2$
ConvBlock3	$32 \times 12 \times 12$	$3\times3, BN, ReLU, 2\times2$
ConvBlock4	$16 \times 6 \times 6$	$3\times3, BN, ReLU, 2\times2$
FC1	512	$BN, ReLU$
FC2	$3 \times M + 5$	None

TABLE I
THE CONFIGURATIONS OF STN ARE SHOWN IN THIS TABLE. THE CONFIGURATION VALUES ARE ARRANGED SEQUENTIALLY AS FOLLOWS: KERNEL SIZE, NORMALIZATION LAYER, ACTIVATION LAYER, AND MAX-POOLING. M REFERS TO THE NUMBER OF LINE SEGMENTS, WHICH WAS SET TO 10.

Table I shows the configurations for the localization network within the STN. While the STN takes an input image with a resolution of 300×300 pixels, the localization network requires an image with a resolution of 100×100 pixels. This implementation is faithful to ASTER STN [2], which also uses down-sampled images for control points prediction.

However, unlike ASTER STN, the localization network here does not output the control points directly. Instead, it predicts 5 polynomial coefficients, as well as $\hat{\phi}$, $\hat{\theta}$, and $\hat{\xi}$ for each of the 10 line segments. Therefore, altogether the localization network's output size is 35 for each image, and these 35 values are converted into 30 control points.

D. Discussion on control points

Figure 3 shows that the rectified images using three control points can better mitigate character-level distortions, and many curved characters from before are straightened when the midpoints are introduced. Moreover with the midpoints, the rectified polynomial forms a straight line across the image, which better aligns with the expected rectified outcome. Therefore, in this work, the use of three control points per line segment is favored, in contrast with similar polynomial scheme [3] which uses only the line segments' endpoints.

Do note that control points predicted outside the image are clamped to the image boundary as per the implementation of ASTER STN, and in that case, the truncated line segment may cause some distortions in the rectified image.

Method	Regular Text			Irregular Text			Avg. Acc
	IIIT	IC13	SVT	IC15	SVTP	CT80	
	3000	1015	647	2077	645	288	7672
DaViT	94.3	95.7	90.9	77.0	83.4	84.7	88.2
SaViT	94.3	95.6	90.4	76.5	83.9	81.3	88.0$_{(-0.2)}$
STN+DaViT	95.1	93.9	92.1	79.6	87.9	86.8	89.6
STN+SaViT	94.5	94.0	92.4	78.5	87.4	86.8	89.0$_{(-0.6)}$

TABLE II

RESULT OF ABLATION STUDY FOR DAVIT, IN TERMS OF TEXT RECOGNITION ACCURACY (IN PERCENTAGES). THE SUBSCRIPT IN THE CELL REPRESENTS ITS RELATIVE ACCURACY WITH RESPECT TO ITS DUAL-AXES COUNTERPART.

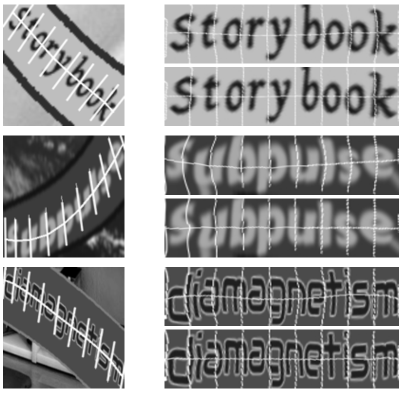

Fig. 3. Rectification results of STN using only the line segment endpoints (top) vs. endpoints + midpoint (bottom).

E. STN training dataset

(a) (b) (c) (d)

Fig. 4. Examples of the 4 types of training data for the STN: (a) curved texts, (b) curved texts with background stripes, (c) curved texts with background stripes & image rotation, and lastly, (d) simple straight text.

The STN training set containing four types of images shown in Figure 4 was generated by the SRNet framework [4]. Customization were made to add a single band of solid-color stripe as the image background. In this paper, $20,000$ samples were generated for each type of training data.

II. DISCUSSION ON DAVIT

In order to study the effectiveness of DaViT, the y-encoder layers were removed and the resulting model is denoted as SaViT. For the training of SaViT, the input image was sliced into $N_x \times P_w H$ strips (as opposed to patches of $P_w P_h$), where $P_w = 2$ and $N_x = 64$. The attention for SaViT only spans across the sequence of strips through the x-encoder layers.

Table II shows the result of the experiments where DaViT generally enhances the overall performance as compared to SaViT with a significant improvement in the irregular test datasets. This suggests that the attention in both axes allows the network to learn a feature representation that is stronger against distortions.

III. EXAMPLES FROM TEST DATASETS

Figures 5,6,7 show the examples of predictions results from the models.

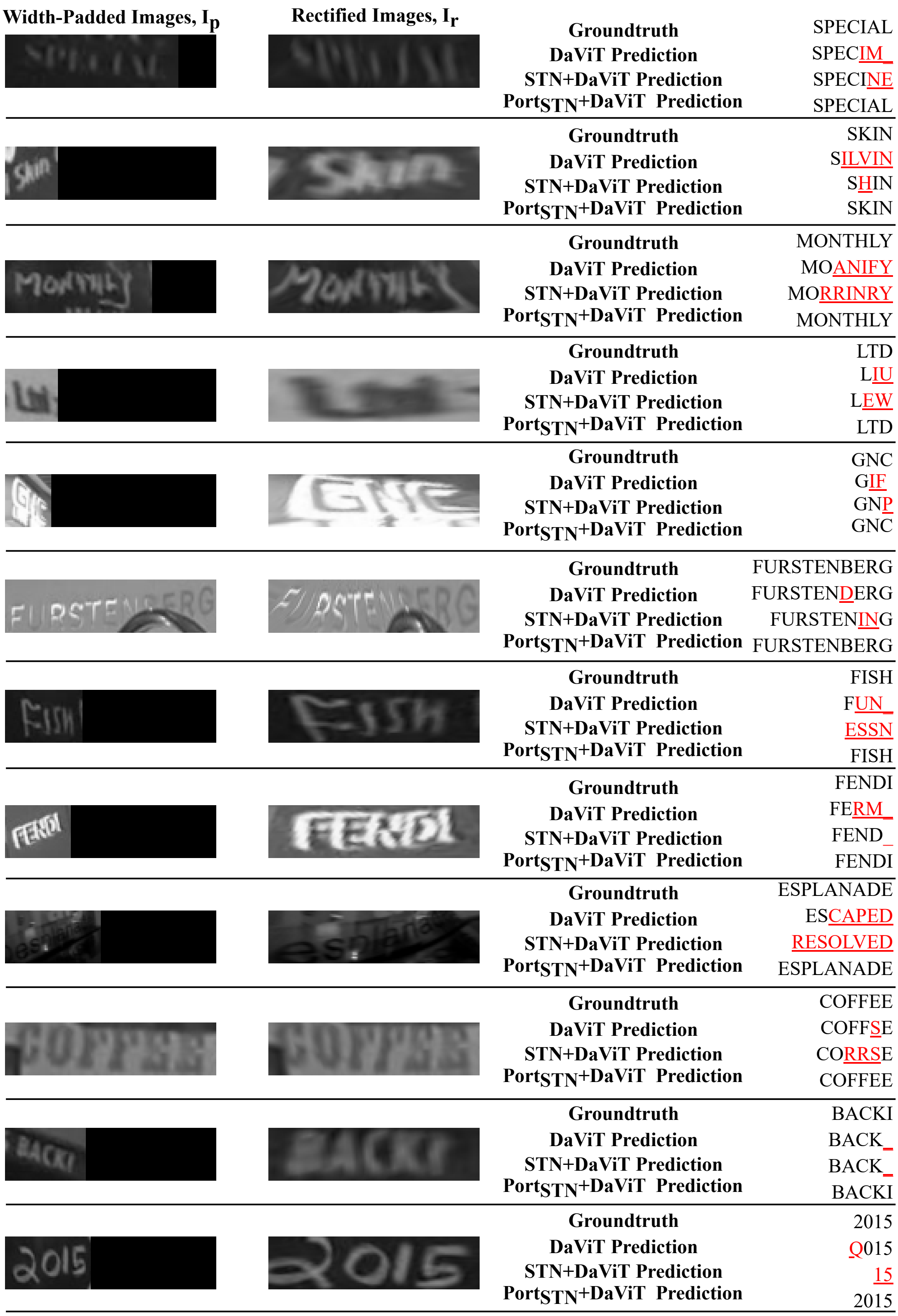

Fig. 5. Examples showing that the combination of failure cases from both DaViT and STN+DaViT, eventually produces accurate predictions when Port$_{STN}$+DaViT was applied instead.

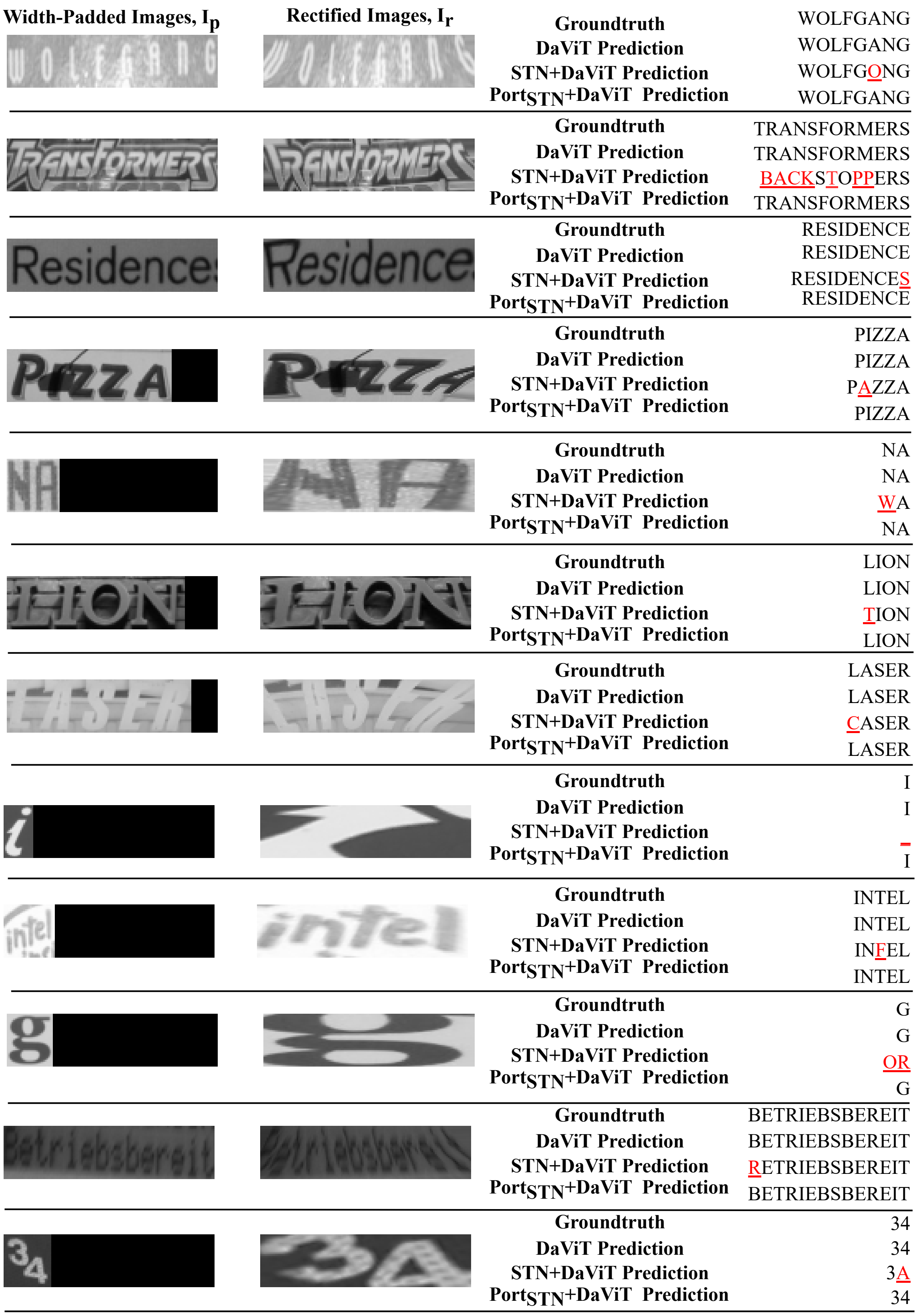

Fig. 6. Examples where correct predictions was achieved by Port$_{STN}$+DaViT mitigating the adverse impacts of rectification.

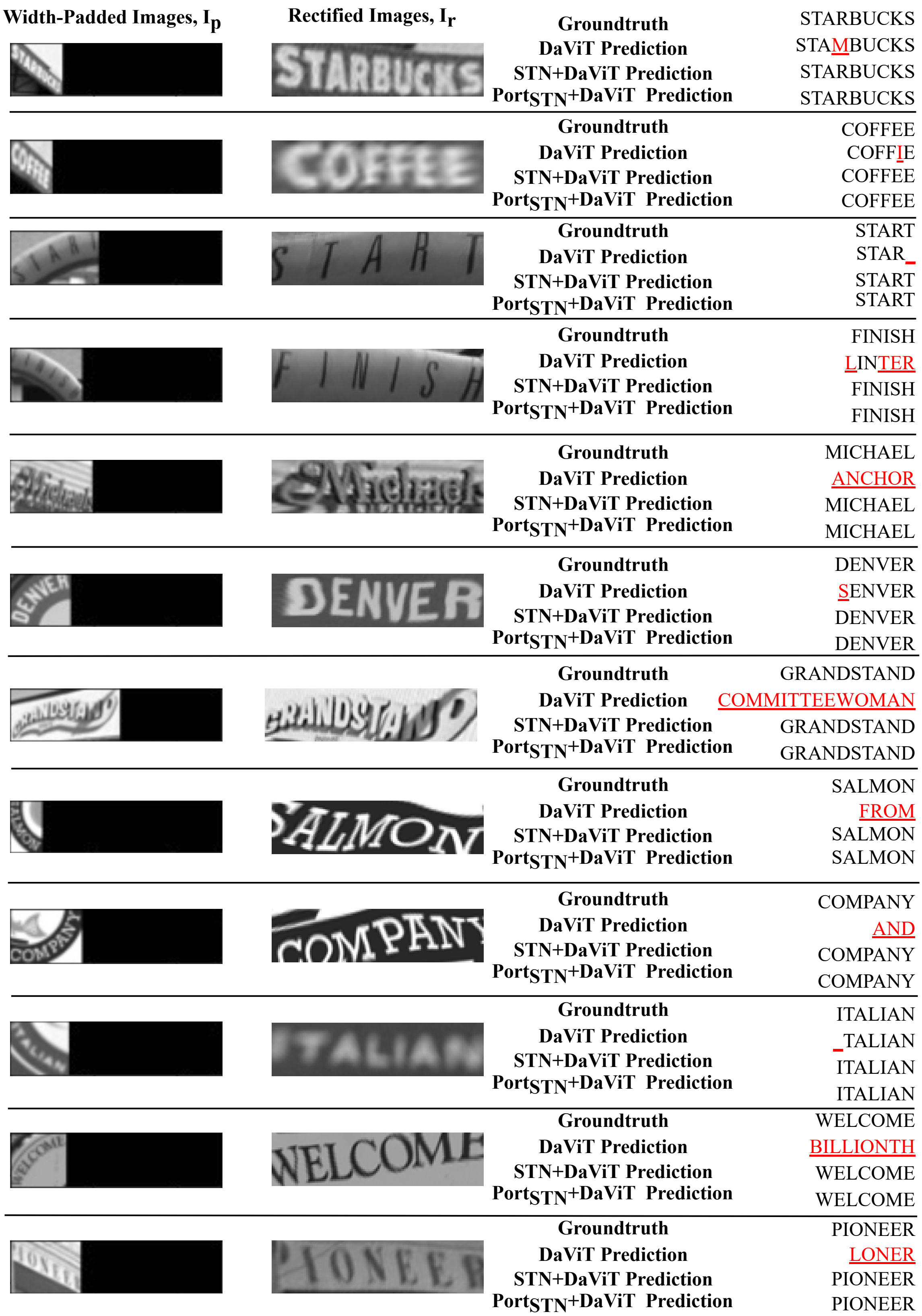

Fig. 7. Examples showing that portmanteau features did not prevent the model from leveraging the benefits of rectification networks.

REFERENCES

[1] N. Gumerov and R. Duraiswami, "Chapter 2 - elementary solutions," in *Fast Multipole Methods for the Helmholtz Equation in Three Dimensions*, ser. Elsevier Series in Electromagnetism, N. Gumerov and R. Duraiswami, Eds. Amsterdam: Elsevier Science, 2004, pp. 39–87.

[2] B. Shi, M. Yang, X. Wang, P. Lyu, C. Yao, and X. Bai, "Aster: An attentional scene text recognizer with flexible rectification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 9, pp. 2035–2048, 2018.

[3] F. Zhan and S. Lu, "Esir: End-to-end scene text recognition via iterative image rectification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 2019.

[4] L. Wu, C. Zhang, J. Liu, J. Han, J. Liu, E. Ding, and X. Bai, "Editing text in the wild," in *Proceedings of the 27th ACM international conference on multimedia*, 2019, pp. 1500–1508.