Searching for Alignment in Face Recognition

Xiaqing Xu1, Qiang Meng1, Yunxiao Qin2, Jianzhu Guo3,4, Chenxu Zhao5, Feng Zhou1, Zhen Lei3,4

1AIBEE, Beijing, China, 2Northwestern Polytechnical University, Xian, China
3CBSR & NLPR, Institute of Automation, Chinese Academy of Sciences, Beijing, China
4School of Artificial Intelligence, University of Chinese Academy of Sciences
5Academy of Sciences, Mininglamp Technology, Beijing, China

Abstract

A standard pipeline of current face recognition frameworks consists of four individual steps: locating a face with a rough bounding box and several fiducial landmarks, aligning the face image using a pre-defined template, extracting representations and comparing. Among them, face detection, landmark detection and representation learning have long been studied and a lot of works have been proposed. As an essential step with a significant impact on recognition performance, the alignment step has attracted little attention. In this paper, we first explore and highlight the effects of different alignment templates on face recognition. Then, for the first time, we try to search for the optimal template automatically. We construct a well-defined searching space by decomposing the template searching into the crop size and vertical shift, and propose an efficient method Face Alignment Policy Search (FAPS). Besides, a well-designed benchmark is proposed to evaluate the searched policy. Experiments on our proposed benchmark validate the effectiveness of our method to improve face recognition performance.

Introduction

Face recognition is a long-standing topic in the research community of computer vision. A standard pipeline of the recognition framework consists of four individual steps: locating faces with bounding boxes and fiducial landmarks, aligning face images using a pre-defined template, extracting face representations and representation comparing. The second step, also named as face alignment (in Fig. 2), serves as deforming face images such that fiducial points are spatially aligned and simplifies the recognition task by normalizing the in-plane rotation, scale and translation variations. However, most recent works (Taigman et al. 2014a; Sun, Wang, and Tang 2014a; Schroff, Kalenichenko, and Philbin 2015; Liu et al. 2017; Deng et al. 2019; Kang et al. 2019) on face recognition focus on designing loss functions and exploring network structures. In contrast, the alignment procedure before model training is less studied.

In this paper, we first explore the effects of the alignment templates (Deng et al. 2019; Zhu et al. 2019; Guo et al. 2020b) on face recognition performance. Face features can be divided into two sets depending on the zone where they are located: internal features, including eyes, nose and mouth, and external features, composed by the hair, chin and face outline. The benefits of external information have been observed in some early works (Lapedriza, Masip, and Vitria 2005; Andrews et al. 2010), but they are rarely discussed in the modern face recognition framework (Taigman et al. 2014a; Schroff, Kalenichenko, and Philbin 2015; Liu et al. 2017; Deng et al. 2019). Significant differences in the 1v1 results are observed by using templates with different degrees of external features involved, as illustrated in Fig. 1. An open problem arises: is there an optimal template such that the produced face region gives the best recognition performance? Specifically, it remains unknown whether fewer backgrounds or irrelevant textures to face (e.g., hair, forehead) benefit face recognition. Besides, it is unclear whether the optimal template generalizes well across various conditions including the pose, age and illumination.

Instead of manually designing templates, we propose to automate the process of finding the optimal template for recognition. To this end, we decompose differences of templates into vertical shift and crop size, and construct a well-defined discrete searching space. We call the vertical shift and crop size pair an alignment policy. The equivalence relation of the alignment policy and the template is described and proved in Section Face Alignment, and illustrated in Fig.
2. The template searching space is thus projected to the cropping box spanned by vertical shift and crop size.

A straightforward way to search for the template is using the grid search. However, grid search is inefficient and costly. For example, the total size of searching space in our work is 93 and the grid search for the optimal template on the dataset like CASIA (Yi et al. 2014) is rather time-consuming (costs about $9102$ GPU hours with 8 Tesla V100 GPUs).

In this paper, we propose an evolution-based method named Face Alignment Policy Search (FAPS) to efficiently searches for the optimal template. FAPS jointly trains a population of models with evolving templates. Inspired by PBT (Jaderberg et al. 2017), we reuse the partially trained weights to accelerate the searching procedure, as training from scratch on a large-scale dataset is time-consuming. To improve the generality of the partially trained model, we set the upper bound of search space as $\text{SuperROI}$ such that the models have the knowledge of all the facial parts and can concentrate on the more informational area. The original explore in PBT mainly considers perturbing the hyperparameter from a better-performing population or resampling new hyperparameter from originally defined distribution, while ignores the relations among different templates in our problem. To accelerate the discovering of better crop size and vertical shift, we propose Intersection based Crossover to combine the strength of well-performing templates (Fig. 5).

Until now, searching for alignment in face recognition is less-studied and there exists no common protocol for evaluation, thus we introduce a well-designed benchmark (including LFW (Huang et al. 2008), AgeDB-30 (Moschoglou et al. 2017) and MultiPIE (Gross et al. 2010), etc.) to evaluate the searched face crop template.

Our main contributions include: (i) To the best of our knowledge, we explore and highlight the effects of alignment templates on face recognition for the first time. (ii) We construct a well-defined searching space by decomposing the template searching into crop size and vertical shift searching, and propose an efficient method named FAPS for template searching. (iii) A well-designed benchmark is proposed to evaluate the searched policy. Extensive experiments on the proposed benchmark validate the efficacy of FAPS.

### Background

**Face Alignment** is used to align faces to a unified distribution and reduce the geometric variations. The most commonly adopted way is applying a 2D affine transformation to calibrate facial landmarks to predefined 2D (Wang et al. 2018; Deng et al. 2019; Wang et al. 2017; Liu et al. 2017) or 3D templates (Taigman et al. 2014b; Guo et al. 2020a).

Besides the affine transformation, some other works learn non-rigid transformations. For example, ReST (Wu et al. 2017) introduces a recursive spatial transformer to learn complex transformation. (Zhou, Cao, and Sun 2018) use local homography transformations estimated via a rectification network to rectify faces. These methods aim for alignment-free through learning alignment jointly with the recognition network in an end-to-end fashion. Despite their achievements, additional computational cost and loss of identity information limit their usage in real-world applications.

Apart from the types of transformation, another critical element of alignment is how to design a proper facial template. Some early works (Lapedriza, Masip, and Vitria 2005; Andrews et al. 2010) have observed performance improvements when including some external face features (i.e., hair, chin and face outline) compared to using internal face features alone (i.e., eyes, nose and mouth). One optimal solution is to apply multi-patches methods (Sun, Wang, and Tang 2014b; Sun et al. 2014; Sun, Wang, and Tang 2014c; Liu et al. 2015) which process an image via multiple templates and dump them to different recognition models. Although this strategy improves performances, it requires too much additional computational costs and carefully designed ensemble methods. In our work, we compare the performance of a set of templates and aim to find the optimal one for the face recognition task.

**Hyperparameter Optimization.** As face alignment policy is a hyperparameter for face recognition, our work closely correlates with the hyperparameter optimization (Feurer and Hutter 2019) problem which automatically tunes the hyperparameters. An RL-based method called AutoAugment (Cubuk et al. 2019) is proposed to train a controller to search for the best data augmentation policy based on specific datasets and models. Apart from the RL-based methods, evolution-based methods (Jaderberg et al. 2017; Ho et al. 2019) spring recently. For example, PBT (Jaderberg et al. 2017) jointly trains a population of models and searches for their hyperparameters with evolution to improve the models’ performances. Exploit and explore are the two most important strategies of PBT. Exploit is responsible for copying weights and hyperparameters from a well-performing model to the inferior one. Explore creates new hyperparameters for the poor-performing model by either resampling new hyperparameters from the originally defined prior distribution or perturbing the copied hyperparameters from a well-performing model. These two strategies make PBT faster and more effective.

In this work, inspired by PBT, we develop a novel evolution-based method named FAPS to search for a better face alignment strategy. The exploit and explore from PBT are also adopted in our method.

### Methodology

In this section, we first review the face alignment process via 2D affine transformations and demonstrate that template searching can be decomposed into searching crop size and vertical shift. Then we detail the proposed FAPS.

**Face Alignment**

We define one alignment template as a composition of landmarks $R_i$ with cropped area $[0, 0, w_b, w_b]$ (a $w_x \times w_y$ rectangle with top left point $[0, 0]$). In this work, facial landmarks in all templates share the same shape. To be more specific, any $R_i$ can be transformed from one base landmarks $R_0$ by scaling $s_i$ and shifting $x_i, y_i$ over the $x, y$ axis respectively as shown in Fig. 2.

One face image $I$ is aligned to landmarks $R_i$ by a 2D affine transformation $T$. Denote $I_n^b$ as the transferred image based on landmarks $R_i$. We seek an optimized affine
Figure 2: An overview of the face alignment process. Assuming we have a template with landmarks \( R_i \) and cropping rectangle from point \((0, 0)\) to point \((w_i, b_i)\). \( R_i \) can be transferred from \( R_0 \) by scaling \( s_t \) and shifting \([x_t, y_t]\), i.e., \( R_i = A_i R_0 \). Then we have \( I_0^i = T^* \{ I \} = A_i T_0^i \), which shows that the transformed image based on landmarks \( R_i \) can be achieved by performing transformation \( A_i \) on \( T_0 \). The final aligned image is the area \([0, 0, w_i, b_i]\) of transferred image \( I_0^i \), which is given by the following steps: 1) Transform image \( I \) to \( I^i \) based on the base landmarks \( R_0 \), 2) Crop the image with area \([x_i, y_i, w_b, s_i, w_b, s_i]\). 3) Resize the area by size \([w_b, b_b]\).

Therefore, instead of designing various templates and aligning a face multiple times, we simplify the processes by aligning once by the base template \( R_0 \) and operating (crop + resize) on the same image \( I_0 \). In our implementation, landmarks in all templates are placed to be horizontally symmetric, which makes \( x_i = 0 \). Let \( m_i = w_b \cdot s_i \), \( \delta_i = y_i / s_i \), our target now is to find the optimal \( m^*, \delta^* \). We call \( p = \{m, \delta\} \) an alignment policy and each policy represents a corresponding template.

Search Space
To facilitate the search process, we place the base face landmarks \( R_0 \) to a 300 \times 300 canvas with the mid-point of the nose (red point in Figure 3(a)) at the center. We denote this template as \( T_0 \). After aligning an image to \( R_0 \), FAPS searches for the optimal region to simulate the effects of applying different templates. A candidate region is determined by 1) crop size \( m \) which controls the tightness of cropped face and 2) vertical shift \( \delta \) which controls the center of cropped area. Some examples are presented in Fig. 3(c).

Denote \( P \) as the union of all candidate \( p \), i.e., the search space. We define the search space as follows: With upper bound \( m_{\text{max}} \) and \( \delta = 0 \), the selected region is able to cover both internal and external face features (Fig. 3(b)). While with \( m_{\text{min}} \) and \( \delta = 0 \), only indispensable facial parts (eyes, nose, mouth) are kept as shown in Fig. 3(c).

Through the variation of vertical shift \( \delta \), some facial features are dropped and some new features are included in the input. When \( m \) is set to the smallest scale \( m_{\text{min}} \), this phenomenon becomes more obvious (Fig. 3(c)). If \( \delta \) is set to the maximum value \( \delta_{\text{max}} \), only the eyebrows are preserved, the forehead is almost omitted. When \( \delta \) is set to the minimum value \( \delta_{\text{min}} \), only the mouth is preserved, the chin is dropped. With such an extreme setting of \( \delta \), the importance of different facial areas can be discovered.

Search Strategy
Denote the recognition model as \( f \) and its weights as \( w \), we represent model trained with images aligned by \( p \) as \( f(w|p) \). Let \( \mathcal{L}_{\text{train}} \) and \( \text{ACC}_{\text{val}} \) be the training loss and validation accuracy, respectively. The process of finding the optimal alignment policy can be formulated as:

\[
p^* = \arg\max_{p \in \mathcal{P}} \text{ACC}_{\text{val}}(f(w^*|p)) \tag{1}
\]

\[
s.t. \ w^* = \arg\min_{w} \mathcal{L}_{\text{train}} f(w|p) \tag{2}
\]

To find the optimal solution, the trivial approach like grid search is to traverse all possible \( p \). In this way, model \( f \) needs to be trained \(|\mathcal{P}| \) times, which is time-consuming and inefficient. Inspired by Population based Training (PBT) (Jaderberg et al. 2017), we train a fixed population of models with different \( p \) in parallel. The “exploit-and-explore” procedure is applied to the worse performing models at a certain interval, where the inferior model clones the weight of better performing model and updates the alignment policy through perturbing this well-performing model’s \( p \). The model can be trained with a new \( p \) without
Intersection based Crossover

SuperROI

To improve the generality of partial trained model when cloning the weights, we initialize \( p \) to \( \{m_{\text{max}}, 0\} \) as shown in Fig. 3 (b), i.e., an initialized Region of Interest (ROI) containing all internal features (eyes, nose and mouth) and external features (jaw-line, ears, part of the hair, etc.). Under this setting, beginning models can have the capacity to handle information from all facial parts. When switching to other policies, the facial region can be a part of the initial one and no new facial parts are introduced. Models only need to learn the trade-offs from current features, i.e., learn to focus on remaining facial parts and ignore removed ones. This process shares the spirit of the supernet in Neural Architecture Search (Chen et al. 2019; Guo et al. 2019; Chu et al. 2019), consequently, we name \( p_0 = \{m_{\text{max}}, 0\} \) as SuperROI.

Intersection based Crossover

The original explore of PBT either re-samples new hyperparameter directly from the originally defined prior distribution or perturbs the current hyperparameter from a well-behaved population to upgrade the weak-behaved population. The former strategy, which resembles random search (Bergstra and Bengio 2012), can relieve the problem of local minima but cannot guarantee qualities of sampled hyperparameters. The later strategy is analogous to the mutation in genetic algorithms and has a high probability of finding better hyperparameter. However, it generates new hyperparameter depending on one particular hyperparameter each time instead of hyperparameters of well-behaved populations, which may lead to unstable results. Besides the above hyperparameter generation methods, the common trend of well-behaved ones is not fully utilized.

Inspired by crossover in genetic algorithms (Spears 1993), we propose Intersection based Crossover to facilitate the discovering of better alignment policy \( p \) during search (Fig. 5). Suppose there exist two well-performing policies \( p_1 = \{m_1, \delta_1\}, p_2 = \{m_2, \delta_2\} \) and the corresponding facial areas are \( A_1, A_2 \) respectively. Their intersection area \( A_{1,2} = A_1 \cap A_2 \) is highly possible to contain rich facial information that benefits face recognition. Policies generated by trivial crossover \( \{m_1, \delta_2\} \) and \( \{m_2, \delta_1\} \) can possibly represent regions that differ a lot from both \( A_1, A_2 \), which therefore fail to cover the intersection area. Instead, Intersection based Crossover finds the policy whose region has the largest similarity with \( A_{1,2} \). Denote \( A(p) \) as the face region represented by policy \( p \) and \( \text{iou}(A(p), A_{1,2}) = \frac{A(p) \cap A_{1,2}}{A(p) + A_{1,2}} \), we update the policy \( p \) and model weights \( w \) by Eq.3 and Eq.4:

\[
p' = \arg\max_{p \in P} \text{iou}(A(p), A_{1,2}) \quad (3)
\]

\[
w' = w_{i^*}, \quad \text{s.t.} \quad i^* = \arg\max_{i \in \{1, 2\}} \text{iou}(A(p'), A_i) \quad (4)
\]

Figure 4: Overview of the proposed FAPS. We first initialize a fixed population of models with SuperROI \( p_0 \). After each epoch, each model’s accuracy \( v \) on the validation set is calculated. If an under-performing model meets requirement1, the Intersection based Crossover will be operated on the model. Then a new alignment policy is generated by combining the policies of two well-performing models. If an inferior model meets requirement2, explore and exploit will be performed. To be more specific, model weights are copied by those of a superior model and new alignment policy is generated by disturbing a superior policy.

Figure 5: Illustration of Intersection based Crossover. \( p_1 \) and \( p_2 \) are alignment policies of two well-performing populations. Their corresponding regions are \( A_1 \) and \( A_2 \), \( A_{1,2} = A_1 \cap A_2 \) represents the shared area (red rectangle). Our Intersection based Crossover finds a policy \( p' \) which has the largest IOU scores with \( A_{1,2} \) (yellow rectangle). As a result, \( p' \) inherits the intersection area. The \( \text{iou} \) function decides whose weight can be cloned to the inferior model. The IOU score of \( A_1 \) and \( A' \) is larger, hence \( w_1 \) is chosen.
Implementation

The alignment template search process is elaborated in Algorithm 1. The details of the main function are below:

**Step:** In each step, we train the model in one epoch through SGD with ArcFace loss (Deng et al. 2019).

**Eval:** We evaluate the current model on our validation set, the verification rate is calculated as the validation accuracy.

**Ready:** A model is ready to go through the exploit-and-explore or Intersection based Crossover process once 1 epoch has elapsed.

**Requirement1:** The model’s validation accuracy $v$ is between the bottom 1/4 and 3/8 of the population.

**Requirement2:** The model’s validation accuracy $v$ is in the bottom 1/4 of the population.

**Exploit:** Get the weight $w$ and alignment policy $p$ of a model that has validation accuracy $v$ in the top 1/4.

**Explore:** See Algorithm 2 for the explore function. For $m$ and $\delta$, we either perturb the original value or uniformly resample them from all possible values.

**Intersection based Crossover:** We choose two well-performing models $f(w_1, p_1)$ and $f(w_2, p_2)$ whose validation accuracies are in the top 1/4 to generate the new alignment policy $p'$. If $p'$ is already deployed by the current models, an extra explore will be applied to $p'$.

Algorithm 1 Face Alignment Policy Search (FAPS).

**Require:** Current policy search space $\mathcal{P}$, SuperROI $p_0 = \{m_{\text{max}}, 0\}$, population size of models $N$.

1: Initialize $N$ models $f(w, p_0)$
2: for each model $f(w, p_1)$ (asynchronously in parallel) do
3: while not end of training do
4:   $w \leftarrow \text{step}(w, p)$ \quad \triangleright current model with policy $p$
5:   $v \leftarrow \text{ACC}_{val}(f(w, p))$ \quad \triangleright evaluation
6: if $\text{ready}(f, v)$ then
7:   check $v$’s performance among all models
8:   if $v$ meets requirement1 then
9:     generate $w', p'$ via Intersection based Crossover
10:    if $p'$ doesn’t exist currently then
11:      $w, p \leftarrow w', p'$
12:     else
13:      $w, p \leftarrow \text{explore}(w', p')$
14:     end
15:     get $w', p'$ through exploit
16:   end
17: update model populations with new $f(w, p)$
18: return $p$ with highest $v$ among training

**Algorithm 2** The FAPS explore function. When revising the alignment policy based on the current one, the change value is amplified by magnitude parameters.

**Require:** current alignment policy $p = \{m, \delta\}$, SuperROI, magnitude parameters $s = \{s_m, s_\delta\}$

1: for param in $p$
2: if random(0, 1) <0.2 then
3: random sample param uniformly from search space
4: else
5: level = [0.1, 2, 3] with probability [0.1, 0.3, 0.3, 0.3]
6: if random(0, 1) <0.5 then
7: param = param – level $\times s_{\text{param}}$
8: else
9: param = param + level $\times s_{\text{param}}$
10: Clip param to stay within SuperROI

Experiments

FAPS Benchmark

To evaluate the influence of different alignment templates and effectiveness of the proposed FAPS, we introduce a well-designed benchmark that includes searching set, training set, validation set and test set. We present our proposed benchmark in Table 1.

The scale of the training dataset is an important factor for face recognition. We separately employ CASIA (Yi et al. 2014) and MS-Celeb-1M (Guo et al. 2016) as middle-scale and large-scale training and searching datasets. For CASIA, we use the full dataset as the searching data and training data. For MS-Celeb-1M, we use MS-Celeb-1M-v1c \(^1\) which remains the completeness of facial images and is highly clean for training. Searching on the MS-Celeb-1M-v1c directly requires too many computational resources. To reduce the searching time, we sample 30000 identities with 30 images per identity from the whole dataset. This subset is named Reduced MS-Celeb-1M-v1c.

Considering different data distributions and characteristics among datasets of the searching set, we enrich the variety of validation set to ensure the generalization of searched policies. The validation set is designed considering the main challenges of face recognition like age, pose and illumination variations. As a result, we build a validation dataset named Cross Challenge in the Wild (CCW), the images are from three datasets in unconstrained environments: LFW (Huang et al. 2008), AgeDB-30 (Moschoglou et al. 2017) and CPLFW (Zheng and Deng 2018).

The test set including LFW, AgeDB-30, CALFW (Zheng, Deng, and Hu 2017), CPLFW, MultiPIE (Gross et al. 2010) and IJB-A (Klare et al. 2015). More details of the benchmark are presented in Appendix.

**Experimental Settings**

We detect the faces by adopting the s3fd detector (Zhang et al. 2017) and localize 68 landmarks via FAN (Bulat and Tzimiropoulos 2017). Images are affined according to the predefined $300 \times 300$ average face template $T_p$, as shown in Fig. 3(a). Faces are cropped and resized with different alignment policies for searching, but with consistent policies for training, validation and testing. The cropped faces are then resized to $112 \times 112$.

The widely used ResNets (He et al. 2016) with embedding structure (Deng et al. 2019) are employed as our recognition networks. The embedding dimension is set to 512. To accelerate the searching process, ResNet18 is adopted as the searching network. ResNet50 is used to train on the training set. ArcFace (Deng et al. 2019) is served as the loss function during searching and training. We implement FAPS with PyTorch (Paske et al. 2019) and Ray Tune (Moritz et al. 2018).

\(^1\)http://trillionpairs.deepglint.com/overview
During searching, the population size of models $N$ is set to 8. The crop size $m_{\text{max}}$ and $m_{\text{min}}$ are set to 232 and 160, respectively. The vertical shift $\delta_{\text{max}}$ and $\delta_{\text{min}}$ are 24 and -32. We set the magnitude parameter of crop size $s_{m}=8$ and the magnitude parameter of vertical shift $s_{\delta}=4$. Under this setting, we have 93 candidates in the template searching space $\mathcal{P}$. More setting details are shown in Appendix.

### Compared Methods

For comparison, we map the widely-used 5-points template presented in ArcFace (Deng et al. 2019) to the pre-defined 300 $\times$ 300 template $T_p$, which results in policy $\mathbf{p} = \{190, -7\}$. Another 25-points alignment template utilized by MFR (Guo et al. 2020b) and works (Zhu et al. 2019; Guo et al. 2018) is mapped to $\{198, -15\}$. We call policy $\{m_{\text{min}}, 0\} = \{160, 0\}$ the TightROI which involves few external face features. SuperROI as well as the aforementioned three policies are treated as compared policies. We further compare the proposed FAPS with the spatial-transform based methods ReST (Wu et al. 2017) and GridFace(Zhou, Cao, and Sun 2018). Fig. 6 shows some aligned faces with different policies. ReST and GridFace coupled alignment with recognition network, they can hardly be mapped into our search space.

### Searching on CASIA

In this section, CASIA is used as the searching and training sets. The corresponding validation/test sets are presented in Table 1. FAPS’s searching process takes 131 GPU hours with 8 Tesla V100 GPUs. As a comparison, the grid search method with ResNet18 takes about 9102 GPU hours. With the searched alignment policy, we train the ResNet50 from scratch for 32 epochs. The learning rate is initialized by 0.1 and divided by 10 at epoch 20 and 28.

Results are summarized in Table 2 and Table 3 (results of the baseline will be discussed in Ablation Study). We denote the searched alignment policy $\text{FAPS}_C(192, 4)$. Obviously, $\text{FAPS}_C(192, 4)$ surpasses the compared policies on all test datasets. For example, on LFW, $\text{FAPS}_C(192, 4)$ outperforms all other policies, especially the TightROI. With the same training dataset, $\text{FAPS}_C(192, 4)$ achieves a 0.45% improvement above ReST. On AgeDB-30 and CALFW, FAPS$_C(192, 4)$ shows significant improvements over the best results from compared policies by 0.78% and 1.15%. As shown in Fig. 6, FAPS$_C(192, 4)$ drops more hair than ArcFace’s and MFR’s but remains more chin. This indicates that hair is not helpful for face recognition with age challenge as people’s hairstyles usually change during their lifetime, while the chin and the outline of chin remain unchanged.

For profile faces, FAPS$_C(192, 4)$ gains improvement over other compared policies on CPLFW, MultiPIE $\pm 75^\circ$ and MultiPIE $\pm 90^\circ$. On the less challenging MultiPIE $\pm 60^\circ$, FAPS$_C(192, 4)$ performs as well as MFR and TightROI.

These results show FAPS’s searched alignment policy gains superiority over handcrafted ones for faces with large pose variations. This mainly because profile faces are aligned to one side of the images (as shown in Fig. 6). Policies with too small crop sizes (e.g., TightROI) filter out useful face features, while large crop sizes (e.g. SuperROI) can bring irrelevant features and background noise. In contrast, our FAPS can find a trade-off and therefore can focus on key features.

![Figure 6: Face images aligned with different templates.](image_url)
Table 3: Rank-1 recognition rates (%) for different poses at different alignment policies on MultiPIE with ResNet50 backbone. FAPS_C(192,4) and FAPS_M(200,4) denote the policies searched on CASIA and Reduced MS-Celeb-1M-v1c, respectively.

<table>
<thead>
<tr>
<th>Training Set</th>
<th>Method</th>
<th>(\pm90^\circ)</th>
<th>(\pm75^\circ)</th>
<th>(\pm60^\circ)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CASIA</td>
<td>ArcFace (190,7)</td>
<td>89.5</td>
<td>97.0</td>
<td>99.3</td>
</tr>
<tr>
<td></td>
<td>MFR (198,15)</td>
<td>91.2</td>
<td>97.7</td>
<td>99.7</td>
</tr>
<tr>
<td></td>
<td>TighROI (160,0)</td>
<td>90.8</td>
<td>97.6</td>
<td>99.7</td>
</tr>
<tr>
<td></td>
<td>SuperROI (232,0)</td>
<td>90.7</td>
<td>97.1</td>
<td>99.3</td>
</tr>
<tr>
<td></td>
<td>baseline (184,4)</td>
<td>90.4</td>
<td>97.5</td>
<td>99.6</td>
</tr>
<tr>
<td></td>
<td>FAPS_C (192,4)</td>
<td>91.7</td>
<td>98.3</td>
<td>99.7</td>
</tr>
<tr>
<td>MS1M</td>
<td>ArcFace (190,7)</td>
<td>70.4</td>
<td>98.8</td>
<td>100.0</td>
</tr>
<tr>
<td></td>
<td>MFR (198,15)</td>
<td>71.9</td>
<td>98.9</td>
<td>100.0</td>
</tr>
<tr>
<td></td>
<td>TighROI (160,0)</td>
<td>68.7</td>
<td>98.4</td>
<td>100.0</td>
</tr>
<tr>
<td></td>
<td>SuperROI (232,0)</td>
<td>70.7</td>
<td>98.0</td>
<td>99.9</td>
</tr>
<tr>
<td></td>
<td>FAPS_C (192,4)</td>
<td>74.6</td>
<td>99.0</td>
<td>100.0</td>
</tr>
<tr>
<td></td>
<td>FAPS_M (200,4)</td>
<td>76.6</td>
<td>98.8</td>
<td>100.0</td>
</tr>
</tbody>
</table>

Table 4: Results on IJB-A with searched policies FAPS_C(192,4) and FAPS_M(200,4). The training set is MS-Celeb-1M-v1c.

<table>
<thead>
<tr>
<th>Method</th>
<th>Verification (@FAR)</th>
<th>Identification @Rank1</th>
<th>Identification @Rank5</th>
</tr>
</thead>
<tbody>
<tr>
<td>GridFace</td>
<td>92.1</td>
<td>83.9</td>
<td>92.9</td>
</tr>
<tr>
<td>ArcFace (190,7)</td>
<td>94.5</td>
<td>87.1</td>
<td>93.1</td>
</tr>
<tr>
<td>MFR (198,15)</td>
<td>94.7</td>
<td>88.6</td>
<td>93.7</td>
</tr>
<tr>
<td>TighROI (160,0)</td>
<td>93.6</td>
<td>82.1</td>
<td>92.4</td>
</tr>
<tr>
<td>SuperROI (232,0)</td>
<td>95.1</td>
<td>87.4</td>
<td>93.7</td>
</tr>
<tr>
<td>FAPS_C (192,4)</td>
<td>94.8</td>
<td>89.7</td>
<td>93.8</td>
</tr>
<tr>
<td>FAPS_M (200,4)</td>
<td>95.1</td>
<td>91.2</td>
<td>94.1</td>
</tr>
</tbody>
</table>

On both CASIA and MS-Celeb-1M-v1c, the searched alignment policies gain better performance. It shows that compared to current human-designed alignment templates, the optimal one can be searched by FAPS to facilitate the face recognition performance. The searched alignment policy can also generalize across different training datasets. Moreover, although the searched alignment policy of MS-Celeb-1M-v1c is different from CASIA’s, the input facial areas decided by the two searched policies are almost overlapped (IOU 0.92). Almost all chin and part of the forehead are kept for both policies. The results show that adding proper external facial features is beneficial to recognition.

Ablation Study
Effectiveness of Intersection based Crossover
We first evaluate Intersection based Crossover, the method we proposed to facilitate the discovering of better alignment policies. To analyze its impact, we search for the CASIA’s alignment policy under the same setting as that in section Searching on CASIA, but without Intersection based Crossover. The searched policy without Intersection based Crossover is named baseline. The results are summarized in Table 2 and 3. The policy FAPS_C(192,4) discovered with Intersection based Crossover shows better results compared to the baseline at almost all test datasets. Specifically, FAPS_C(192,4) outperforms baseline by 1.0% at CALFW, 1.3% and 0.8% at MultiPIE ±90° and ±75°. At CPLFW, FAPS_C(192,4) is slightly inferior to baseline. The reason may be CPLFW has more background noise and occlusion than MultiPIE. The facial area decided by FAPS_C(192,4) is a bit larger than baseline’s, which means more noise is involved.

Conclusions
In this paper, we explore the effects of different alignment templates on face recognition and propose a fast and effective alignment policy search method named FAPS. The searched templates via FAPS achieve better recognition performance compared to human-designed ones on multiple test datasets and generalize across different training datasets. Besides, our searched templates reveal that except for the internal facial features like eyes, nose and mouth, external features like chin and jawline are helpful for face recognition. This also sheds some light on the further development of face recognition.
Acknowledgments
This work was supported in part by the National Key Research & Development Program (No. 2020AAA0140002), Chinese National Natural Science Foundation Projects #61876178, #61806196, #61976229.

References
Bulat, A.; and Tzimiropoulos, G. 2017. How far are we from solving the 2D & 3D Face Alignment problem? (and a dataset of 230,000 3D facial landmarks). In International Conference on Computer Vision.


