





中國人民大學
RENMIN UNIVERSITY OF CHINA

# Fooling Image Search Engine

Yuanjun Gong, Bin Liang*, Jianjun Huang

{Gongyuanjun, liangb, hjj}@ruc.edu.cn

Renmin University of China

# Outline

1. Introduction
2. Technical Background
3. Methodology
4. Attacking Google Image Search Engine
5. Discussion
6. Conclusion

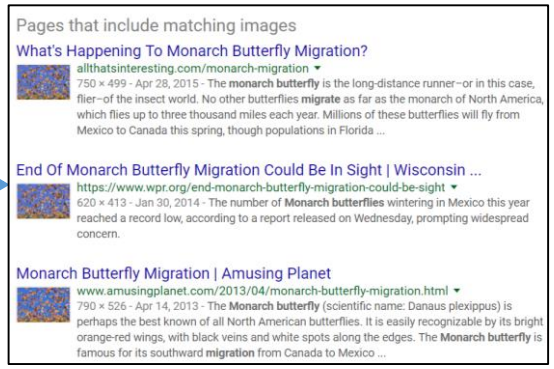
# 1. Introduction

- Image search engines (e.g. Google, Baidu) provide service of reverse image search, or search by image, to allow users to search for related images by uploading an image or image URL.
- **Reverse image search** is a **content-based image retrieval (CBIR)** query technique that involves providing the CBIR system with a sample image that it will then base its search upon; in terms of information retrieval, the sample image is what formulates a search query.[1]



Query image $x$

Google Images




Search result

Reverse image search may be used to

- Locate the source of an image
- Find higher resolution versions
- Track down the content creator
- **Detect plagiarism**

[1]https://en.wikipedia.org/wiki/Reverse_image_search

Google Search

https://www.google.com.hk/search?tbs=sbi:AMhZZiuYF5M78vs4RYpDvcqmCt7WyRVow90vWD5IaXDLnYPqpvmqeLWxl9rXJZYW7Q86-mEZwjH...

应用 图片 其他书签

Google im76.jpg petal Sign in

All Images Maps More Settings Tools SafeSearch on

About 5 results (0.58 seconds)



Image size:
500 × 334

No other sizes of this image found.

Best guess for this image: ***petal***




Query image

Visually similar images



Report images

Pages that include matching images

Climbing Hydrangea Problems | Garden Guides

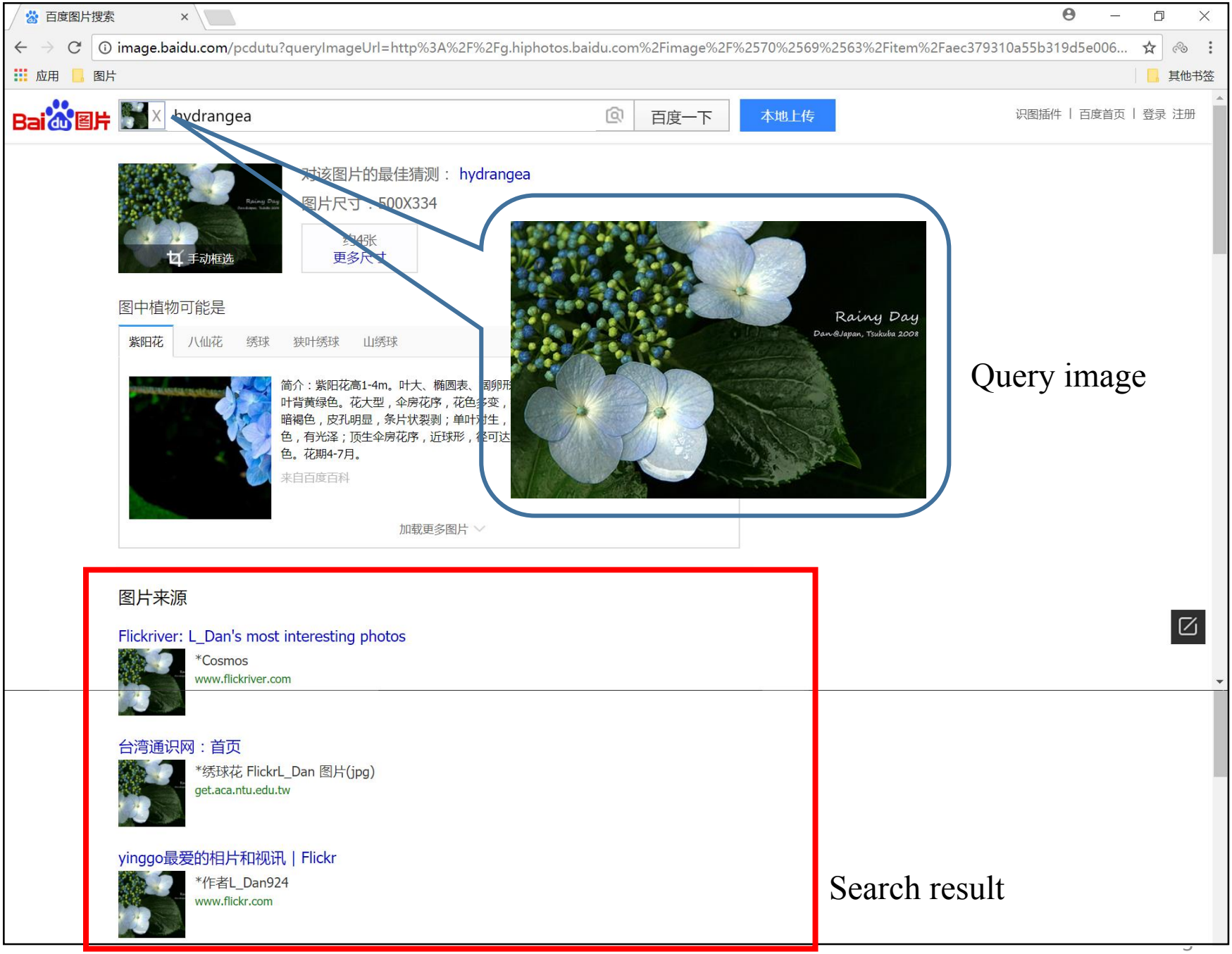
https://www.gardenguides.com › Flowers

500 × 334 - Sep 21, 2017 - The climbing hydrangea ("Hydrangea anomala petiolaris") has many similarities to the upright shrub nydrangeas. The main differences are leaf size and growth habit. The leaves of climbing hydrangea are round, shiny and about 3 inches wide. They will turn an attractive yellow in the fall. The white flowers ...

L_Dan's most interesting Flickr photos | Picssr

picssr.com/photos/minicooper_dan/interesting?nsid=56758835@N00

Search result


百度图片搜索
image.baidu.com/pcdutu?queryImageUrl=http%3A%2F%2Fg.hiphotos.baidu.com%2Fimage%2F%2570%2569%2563%2Fitem%2Faec379310a55b319d5e006...
应用
图片
其他书签
Bai du 图片
hydrangea
百度一下
本地上传
识图插件 | 百度首页 | 登录 注册
对该图片的最佳猜测： hydrangea
图片尺寸：500X334
约4张
更多尺寸
手动框选
图中植物可能是
紫阳花
八仙花
绣球
狭叶绣球
山绣球
简介：紫阳花高1-4m。叶大、椭圆表、
叶背黄绿色。花大型，伞房花序，花色
暗褐色，皮孔明显，条片状裂剥；单叶
色，有光泽；顶生伞房花序，近球形，
色。花期4-7月。
来自百度百科
加载更多图片
Rainy Day
Dan@Japan, Tsukuba 2008
Query image
图片来源
Flickriver: L_Dan's most interesting photos
*Cosmos
www.flickriver.com
台湾通识网：首页
*绣球花 FlickrL_Dan 图片(jpg)
get.aca.ntu.edu.tw
yinggo最爱的相片和视讯 | Flickr
*作者L_Dan924
www.flickr.com
Search result

# Plagiarism Detection

The poster of TV play *Fuyao* released in June 2016, *was* accused of plagiarism. The original image is created in 2014 by *LuHe*, a painter.



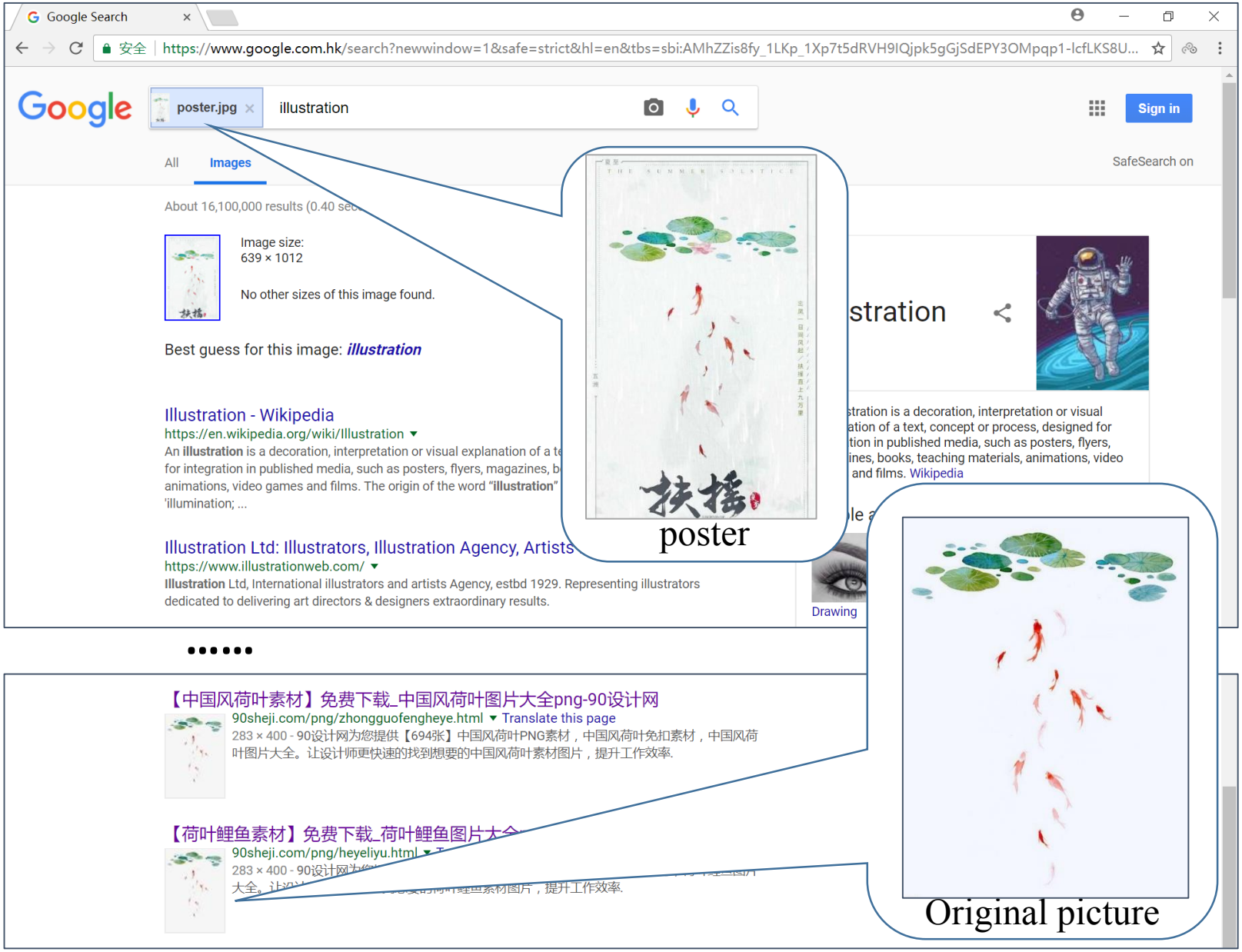
Original image




Poster of a TV play

suspected plagiarism

# Plagiarism Detection



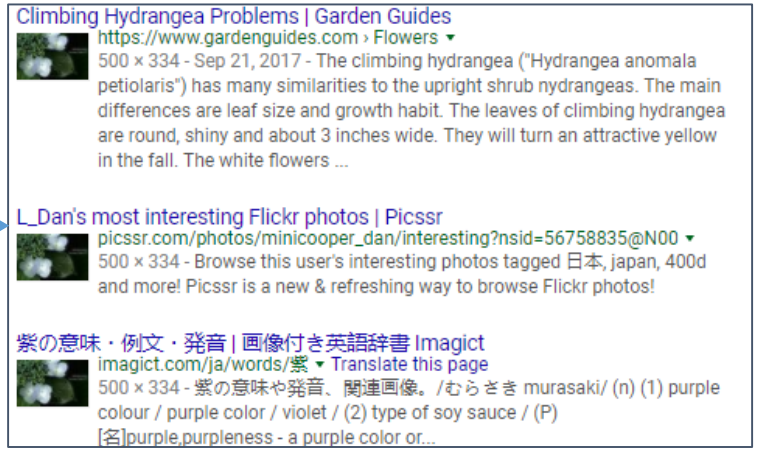
# Threat model

- CBIR system can be a potential attack target.
- Adversaries may evade the search engine by **introducing perturbations to the query image**.




Query image $x$

Google Images




Matching Images




Query image $x$

$+\varepsilon$



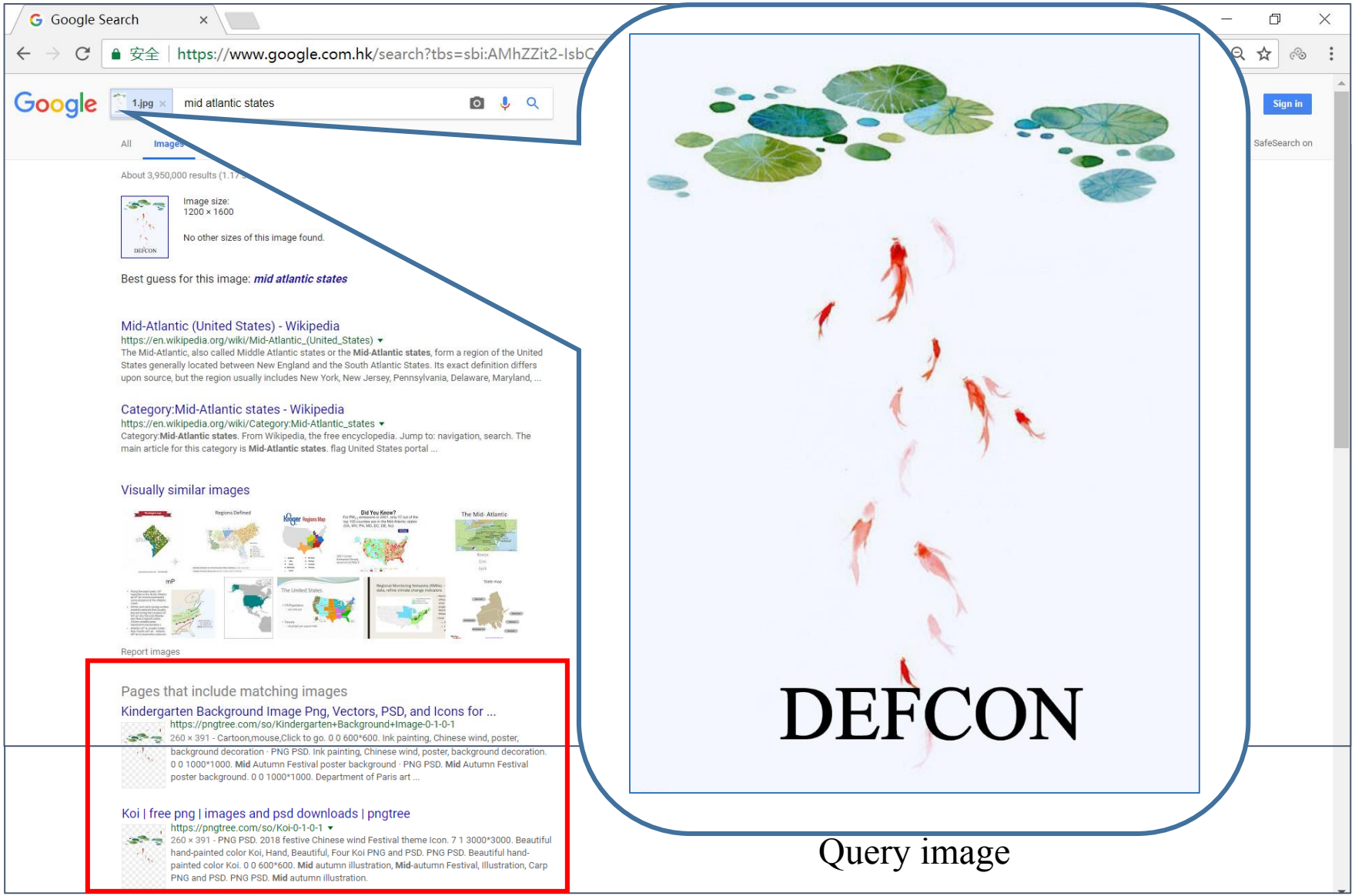

adversarial image $x'$

Google Images

No results

Evasion

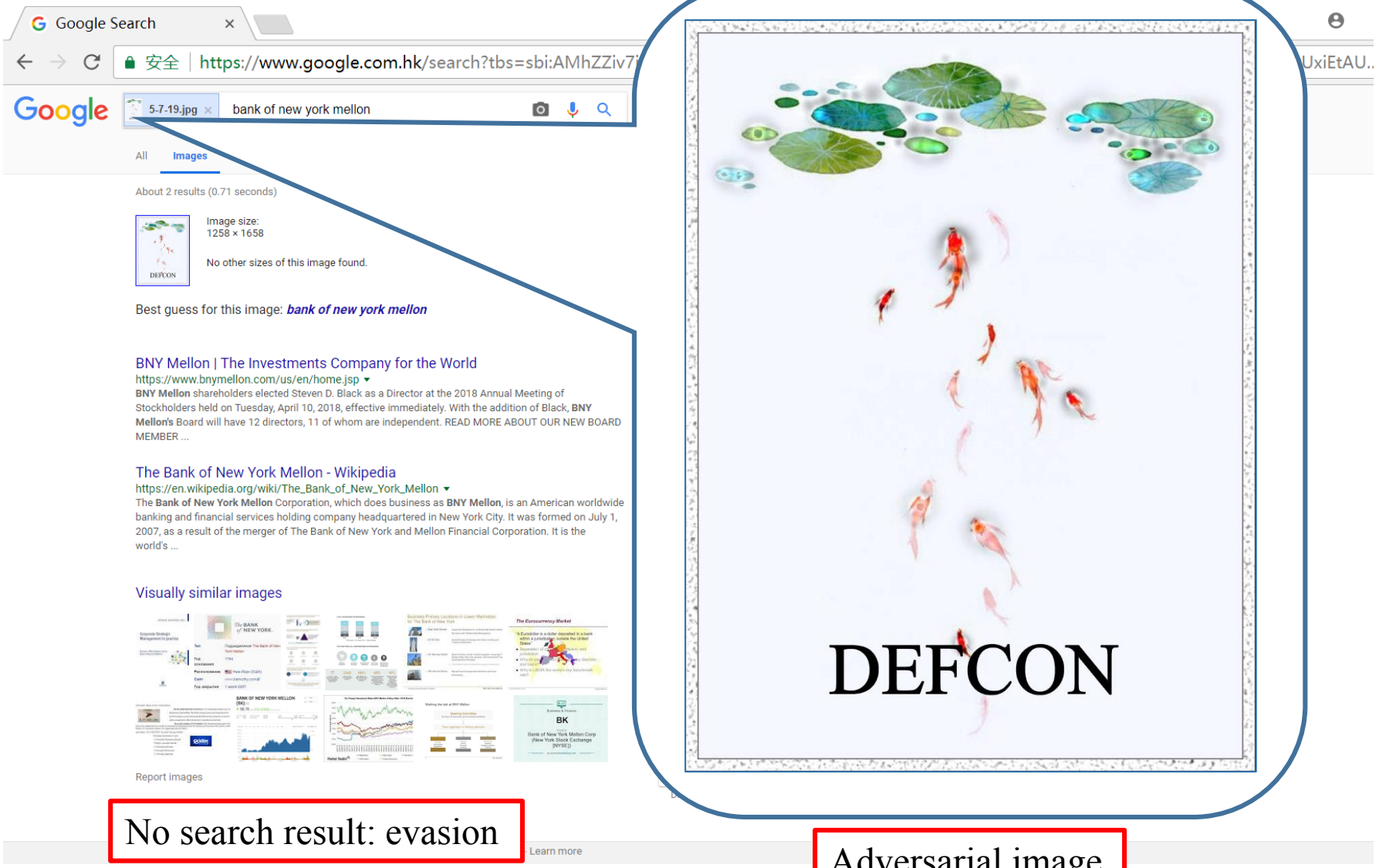
# Evasion




Query image

Search result: original image

# Evasion

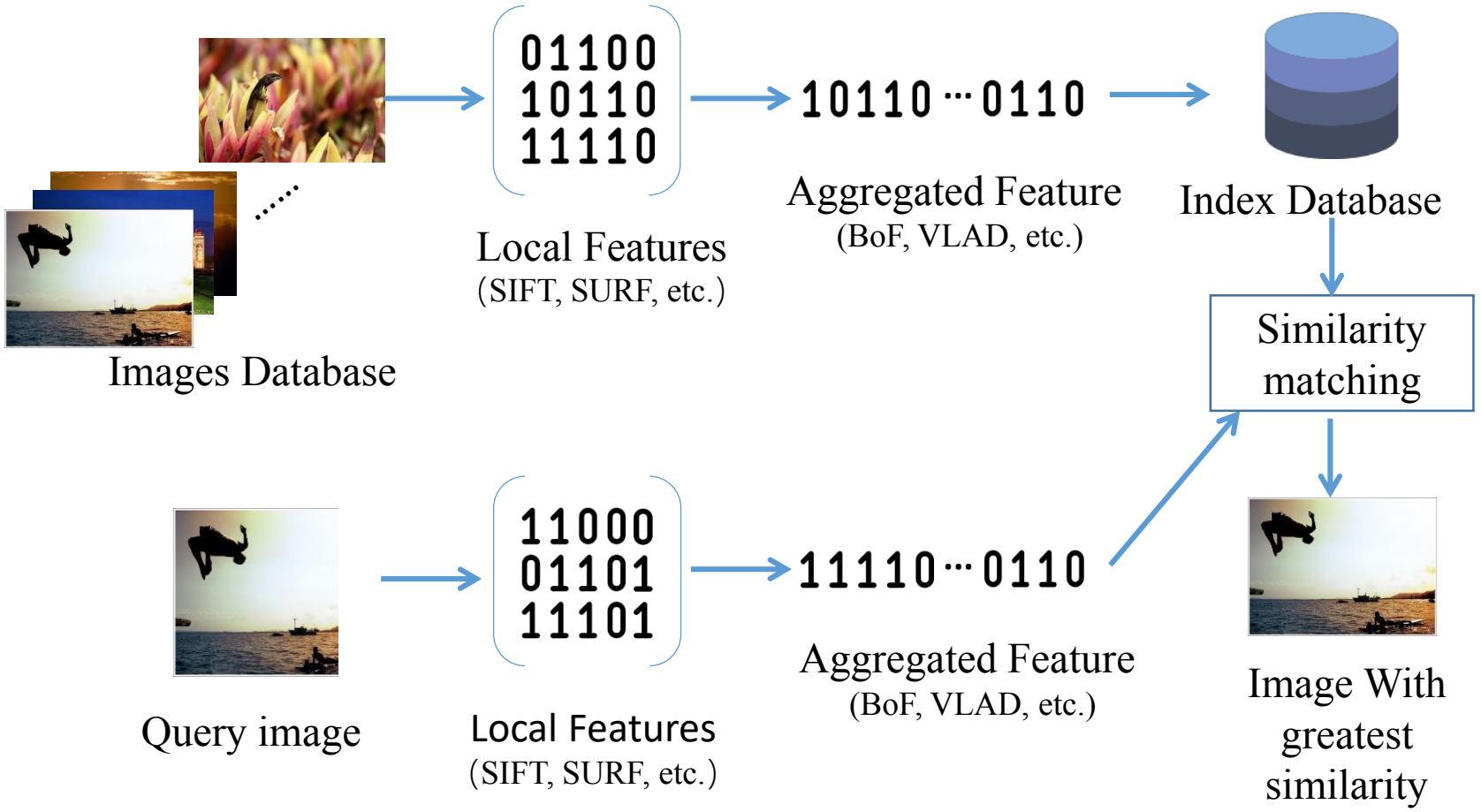



No search result: evasion

Adversarial image

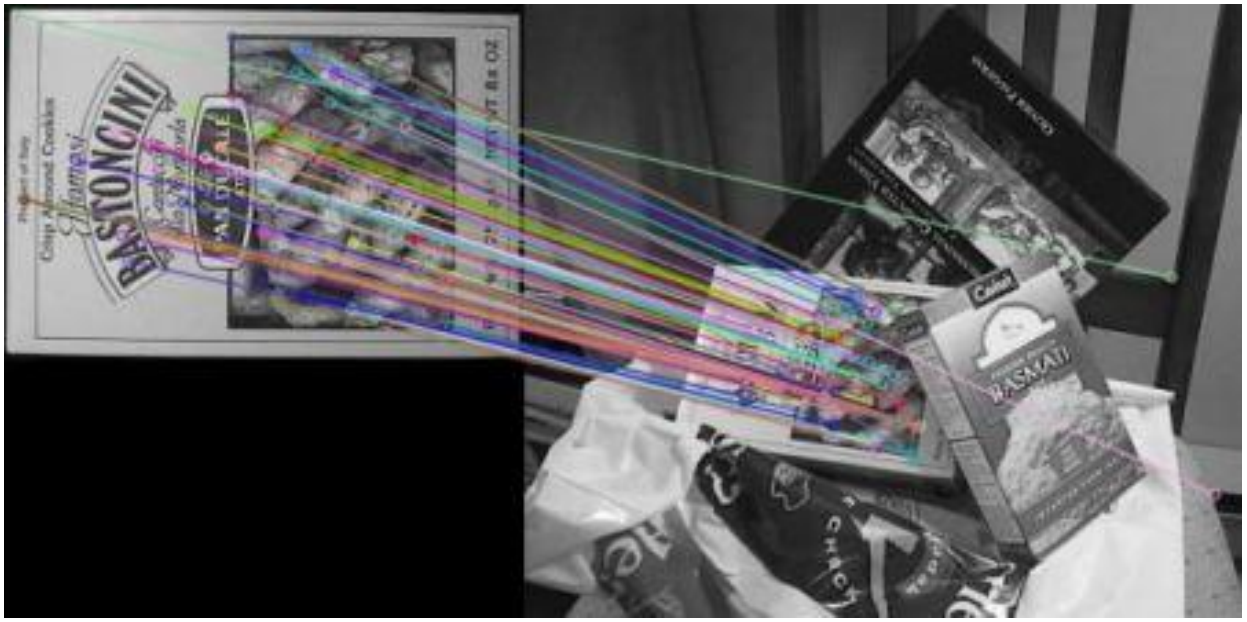
# 2. Technical Background

## CBIR framework



# Local Features

- Essentially, searching similar images in CBIR systems is a image descriptors matching process. **SIFT** (Scale Invariant Feature Transform ) [2] and **SURF** (Speeded Up Robust Features) [3] are widely used local featuring algorithms.




SIFT descriptor matching

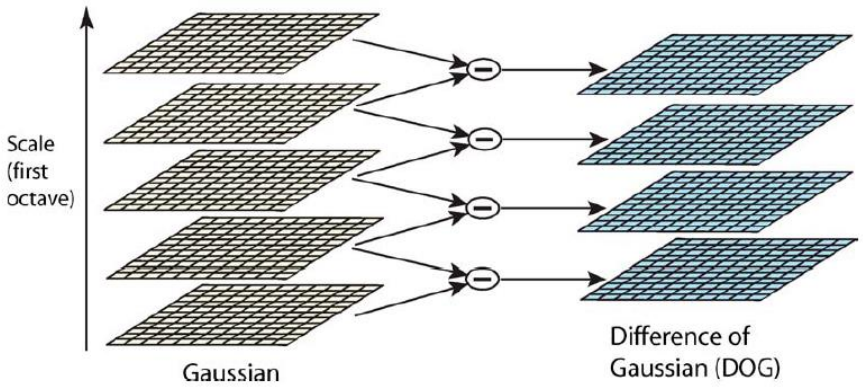
- SIFT is more accurate while SURF has a higher speed.

[1] Lowe, David G., and D. G. Lowe. "Distinctive Image Features from Scale-Invariant Keypoints." International Journal of Computer Vision 60.2(2004):91-110

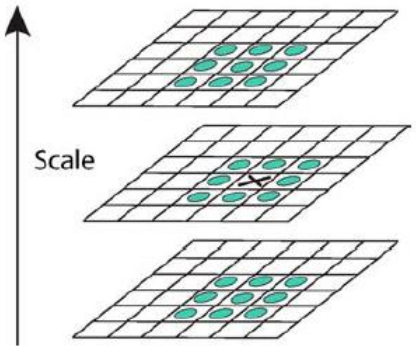
[2] Bay, Herbert, T. Tuytelaars, and L. V. Gool. "SURF: Speeded Up Robust Features." European Conference on Computer Vision Springer-Verlag, 2006:404-417.

# SIFT [2]

SIFT and SURF algorithms both build image pyramids, referred to as scale space, to find keypoints in different scale. They both apply a non-maximum suppression in a 3×3×3 neighborhood to find the potential keypoints. Point $\boldsymbol{x}$ is recorded as a keypoint only when $abs(value(x))$>T, while T refers to the threshold of the algorithm.




Scale space




26 neighbors in 3×3×3 regions

For each octave of scale space, the initial image is repeatedly convolved with Gaussians to produce the set of scale space. Adjacent Gaussian images are subtracted to produce the difference-of-Gaussian images.

# SURF [3]

Instead of using DOG-value, SURF uses the approximation of the determinant of Hessian matrix. Given a point $\boldsymbol{x} = (x, y)$ in an image I, the Hessian matrix $H(\boldsymbol{x}, \sigma)$ in $\boldsymbol{x}$ at scale $\sigma$ is defined as follows

$$H(\boldsymbol{x}, \sigma) = \begin{bmatrix} L_{xx}(\boldsymbol{x}, \sigma) & L_{xy}(\boldsymbol{x}, \sigma) \\ L_{xy}(\boldsymbol{x}, \sigma) & L_{yy}(\boldsymbol{x}, \sigma) \end{bmatrix}$$

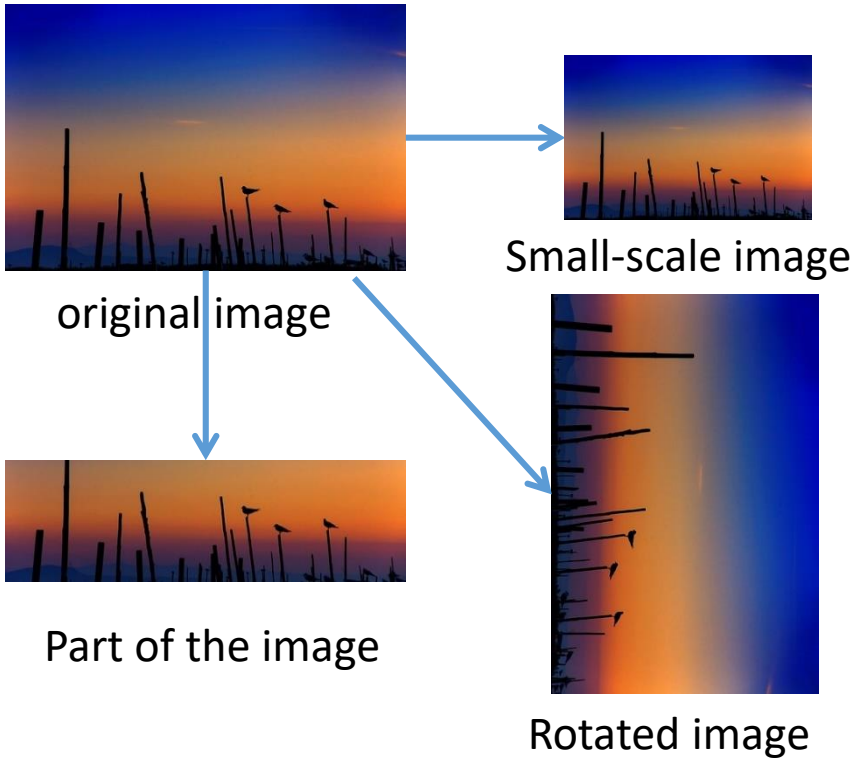
where $Lxx(x, \sigma)$ is the convolution of the Gaussian second order derivative with the image I in point $x$.

Using box filter instead of Gaussian second order derivative to speed up, the approximation of $L_{xx}(\boldsymbol{x}, \sigma)$ is denoted by $D_{xx}(\boldsymbol{x}, \sigma)$.

Therefore $det(H_{approx}) = D_{xx}D_{xy} - (0.9D_{xy})^2$

# Performance

- CBIR systems perform well with rotated image, gray scale image, or even part of the image.




SIFT/SURF are rotation invariable and scale invariable, therefore original image can be retrieved using rotated image and small-scale image as query images.

SIFT/SURF's robustness makes it possible to achieve a successful search with only about 10% keypoints.

# 3. Methodology

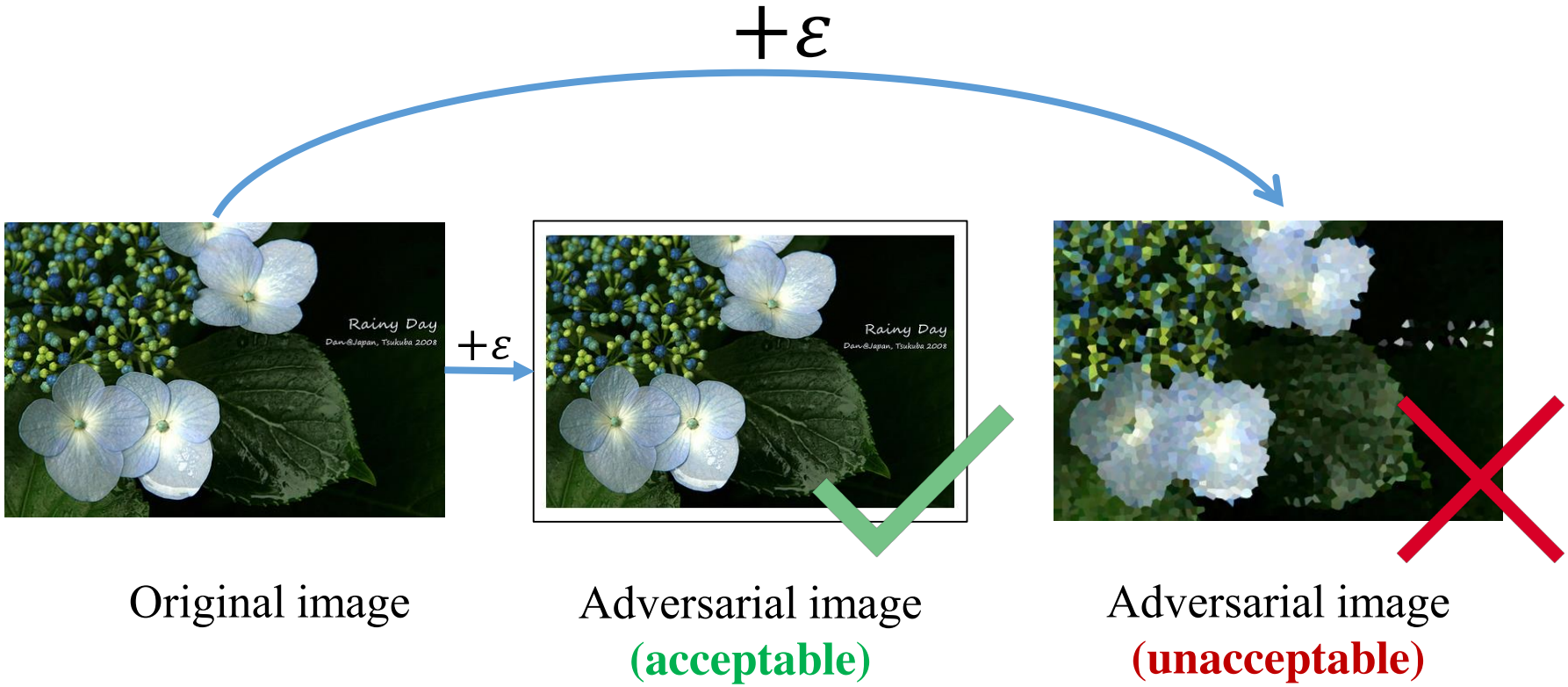
- Descriptor plays an important role in image matching. Therefore, the basic idea of **bypassing an image retrieval system** is to *change the statistical histogram of keypoints descriptors*.

- Two methods
  ① **Removing** original keypoints from the image
  ② **Injecting** external keypoints to the image

- Note that utility-preserving is required.

# Utility-Preserving Requirement

- An adversarial image should be utility-preserving, i.e., it must keep its original visual semantics for a human observer.




Original image

Adversarial image **(acceptable)**

Adversarial image **(unacceptable)**

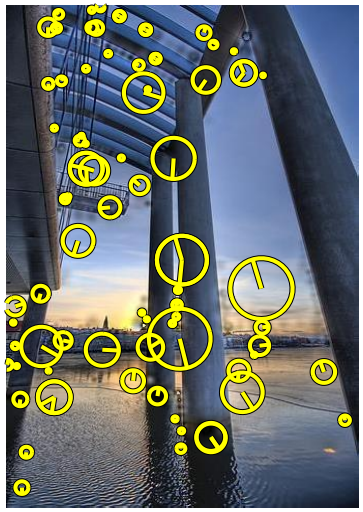
# 3.1 SIFT Keypoints Removing Method

Some SIFT keypoints removal algorithm have been proposed in resent years, We employ the fast and effective RMD [4] method to remove SIFT keypoints.



Original image
187 SIFT keypoints

RMD



Processed image
73 SIFT keypoints

Removal with Minimum local Distortion attack (RMD) targets a limited number of keypoints to be erased. It introduce a value $\delta^+ > 0$ that defines the subset $\varepsilon_{\delta^+} = \{x: C < D(x) < C + \delta^+\}$. Erasing keypoint x in $\varepsilon_{\delta^+}$ means to decrease the absolute of DOG-value |D(x)| by an amount $|\delta|$ such that its new value is below the threshold C ,C is the fixed contrast threshold.

[4] Do, Thanh Toan, et al. "Deluding image recognition in sift-based cbir systems." ACM Workshop on Multimedia in Forensics, Security and Intelligence ACM, 2010:7-12.

# 3.2 R-SURF:SURF Keypoints Removing Method

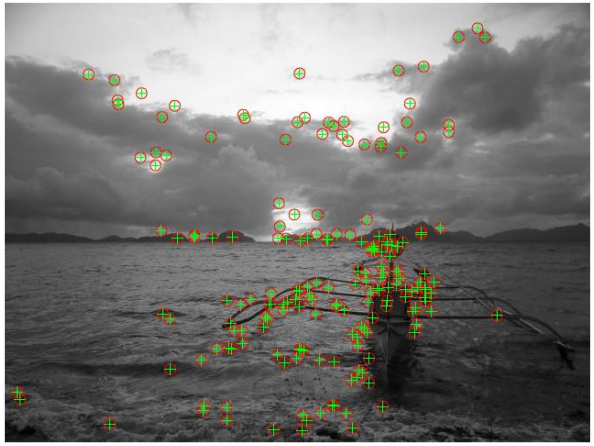
As far as we know, there are no particular SURF keypoint removal algorithms in the early works. Therefore we propose a method called R-SURF to remove SURF keypoints, it is a optimization-based keypoints removing algorithm.

For every target keypoint, support area S(x) is determined and three box filters $K_{xx}$, $K_{xy}$, $K_{yy}$ are reconstructed based on the scale $\sigma$ of the keypoint, and the following nonlinear optimization is constructed.

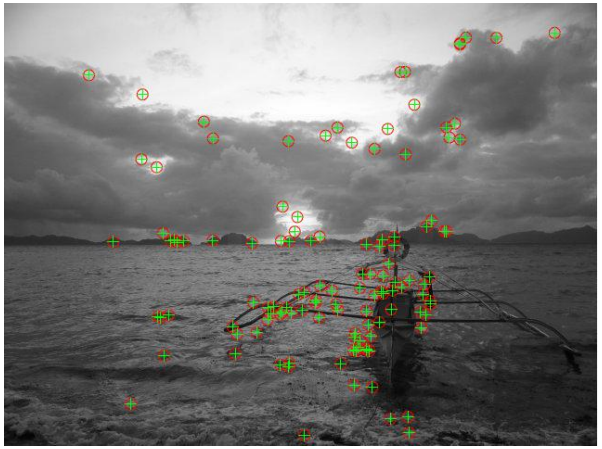
$$\min f = \sum_{(u,v)\in S(\boldsymbol{x})} |\varepsilon(u,v)|$$

$$s.t. \begin{cases} \left(K_{xx}\otimes \mathrm{I}'(\boldsymbol{x})\right) \times \left(K_{yy}\otimes \mathrm{I}'(\boldsymbol{x})\right) - 0.81 \times \left(K_{xy}\otimes \mathrm{I}'(\boldsymbol{x})\right)^2 < T \\ \mathrm{I}(u,v) + \varepsilon(u,v) \geq 0\ , (u,v) \in S(\boldsymbol{x}) \\ \mathrm{I}(u,v) + \varepsilon(u,v) \leq 255\ , (u,v) \in S(\boldsymbol{x}) \end{cases}$$

Where $I'(u,v) = I(u,v) + \varepsilon(u,v)$ , the increment $\varepsilon(u,v)$ is added to each point I(u,v) in support area S(x). Try to find smallest distortion $\sum \varepsilon(u,v)$ when $D'(x,\sigma) < T$ holds.



R-SURF



Original image
186 SURF keypoints

Processed image
122 SURF keypoints

- This algorithm causes very little distortion to image. In general, we only remove some low-level, small-scale keypoints in the images to avoid obvious distortion.

# 3.3 Evaluation: Removal

- Target system:

  The target system **VisualIndex** is a MATLAB and VLFeat based simple image indexing engine, created by Andrea Vedaldi.
  https://github.com/vedaldi/visualindex

  (codebook=10000, image database=flikr2500)



Query image

Query in target system

**VisualIndex**

RMD with **ten** iterations

**Can bypass**



Search result

The system can be bypassed with removal algorithm, when query images contain small amount of keypoints.

# Evaluation: Removal



Query image

Query in target system

VisualIndex

RMD with **ten** iterations

**Cannot bypass**



Search result



Query image

Query in target system

VisualIndex

RMD with **sixty** iterations

**Can bypass**

**But cannot preserve utility**



Search result

# 3.4 Keypoints Injection Method

In order to meet the utility-preserving requirement, keypoints injection methods are introduced as well. Two problems are needed to be addressed.

- Where to inject SIFT keypoints?
  - Inject inside the original image directly
  - Inject to a frame surrounding the image

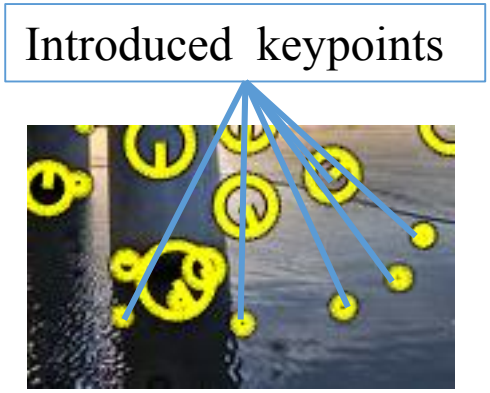
- How to generate SIFT keypoints to be injected?
  - Generate SIFT keypoints with **IMD algorithm** (the inverse operation of RMD)
  - Construct a frame with malformed **basic bricks**

# IMD Algorithm

**IMD** (Injection with Minimum local Distortion ) is the inverse operation of RMD algorithm. It targets a collection of potential keypoints, assigning location and scale information for each point randomly.
For each potential keypoint, the absolute of DOG-value |D(x)| is calculated. If |D(x)| is smaller than the threshold C, a patch will be added upon the original image to meet the threshold. The size and value of the patch is calculated using scale information and Gaussian filter.



Keypoints of original image




Keypoints of resultant image



resultant image

# Inject to a frame

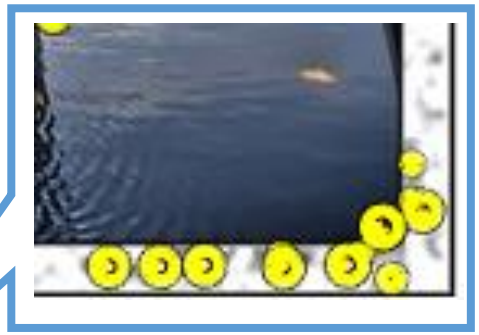


Original image

IMD



IMD image



Introduced keypoints

# Basic Bricks

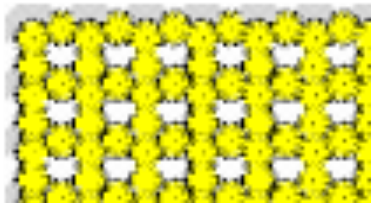
To minimize the size of frame, we construct the frame with basic bricks to generate keypoints as many as possible.
In order to know the visual feature of SIFT keypoints, we extract SIFT keypoints as well as their surrounding regions from images, using K-means clustering algorithm to obtain multiple centroids, and finally build grains based on the result of clustering algorithm. We can see that a keypoint generates where there is an edge or corner.




One cluster of k-means



grain



Keypoints visualization

A kind of grain is build. The contrast of grain should be changed to meet different SIFT threshold. Keypoints are generated as many as possible.

# 3.5 Evaluation: Injection



Query in target system

**VisualIndex**

IMD inside image

**Can bypass**



Query image

Search result

The system can be bypassed with injection algorithm, when query images contain small amount of keypoints.

# Evaluation: Injection



Query image

Query in target system

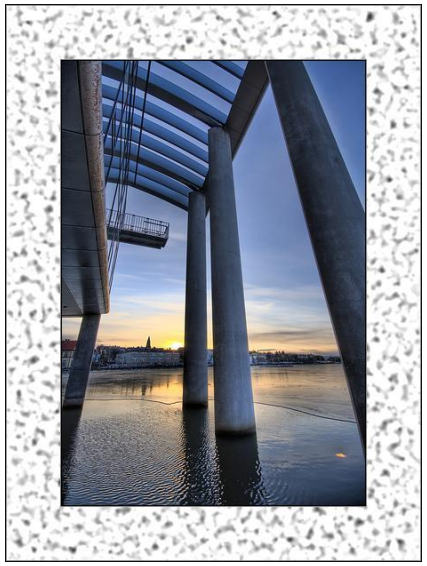
**VisualIndex**

IMD inside image

**Cannot bypass**



Search result



Query image

Query in target system

**VisualIndex**

Inject frame (IMD)

**Cannot bypass**



Search result

# 3.6 Evaluation: Hybrid



Query image

Query in target system

**VisualIndex**

RMD removal + IMD Injection



Search result



Query image

Query in target system

**VisualIndex**

RMD removal + basic bricks Injection



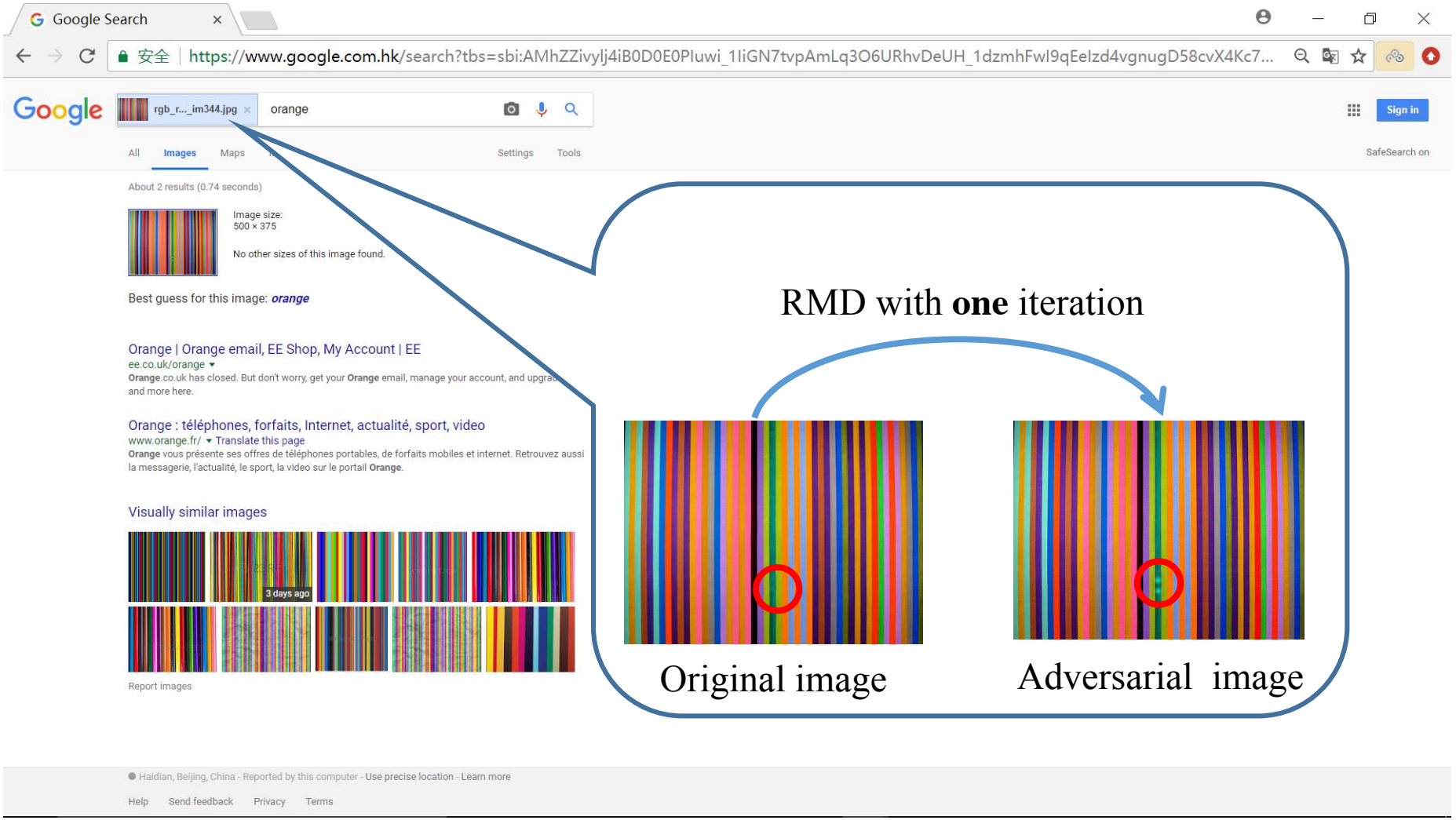
Search result

This experiment shows that it is feasible to bypass VisualIndex system with removal and injection algorithms.

# 4. Attacking Google Image Search Engine

- Challenge
  - Unknown algorithm
  - 0/1 feedback
- Our guess
  - Google image retrieval system uses local descriptors such as SIFT/SURF.
- Bypassing strategy
  - Removal only
  - Injection only
  - Hybrid: Combination of the above two methods

# Removal Only




For some images that contain very few keypoints, performing RMD with one iteration can bypass google image search engine successfully.

# Removal Only




Original image




RMD with **one** iteration
**CANNOT** bypass



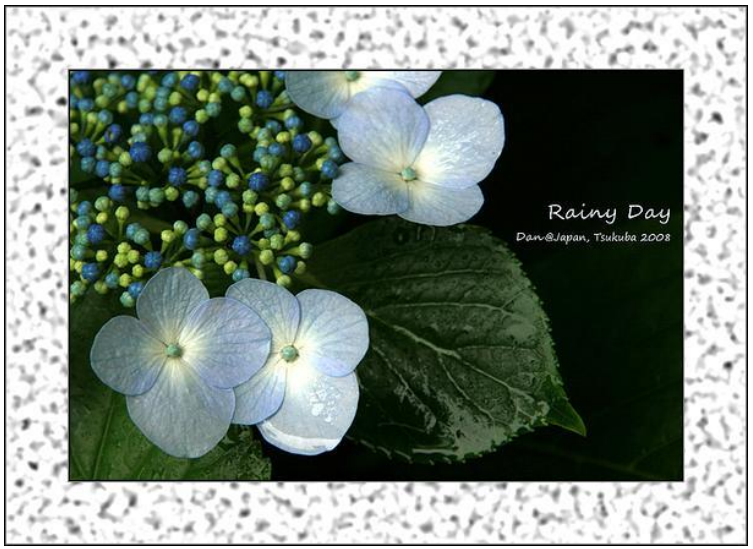

RMD with **five** iterations
**CAN** bypass

Keypoints removal algorithms may cannot remove big-scale keypoints without making unacceptable distortion. Experiment shows that if leaving the big-scale keypoints unremoved, 10% of keypoints can lead to correct searching results, due to the robustness of CBIR system.

# Injection Only



IMD on **15-pixel-wide** frame
**CAN** bypass Google image search




IMD on **50-pixel-wide** frame
**CANNOT** bypass Google image search

Injection only strategy performs well on some simple images, however, as the image on the right shows, sometimes 'injection only' strategy is useless and unacceptable. On account of *utility preserving*, we should make the adversarial image as natural as possible. Strategies '*Removal only*' and '*Injection only*' may not meet this requirement sometimes. We should use a hybrid strategy for better performance.

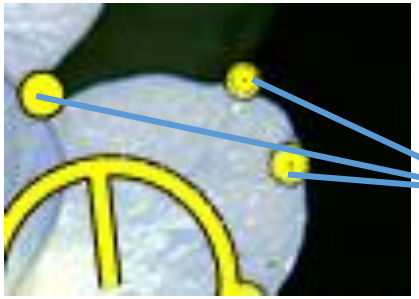
# Hybrid: Combination of Removal and Injection






RMD with **one** iteration + IMD on **15-pixel-wide** frame

SIFT keypoints visualization

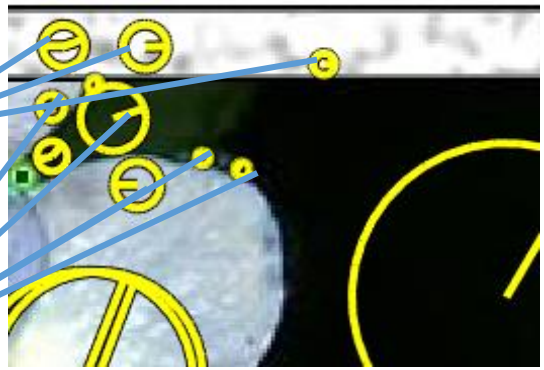


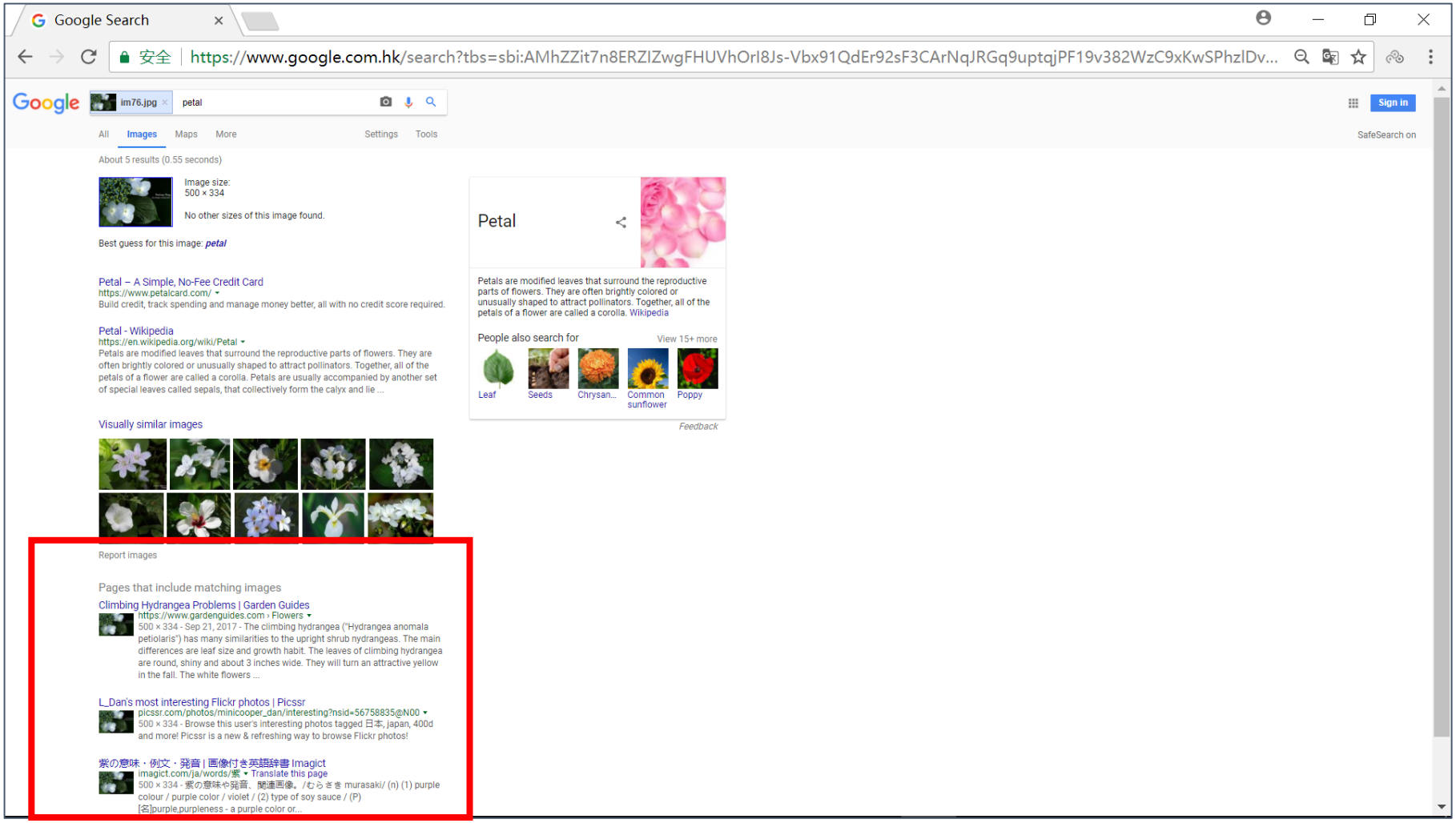
SIFT keypoints of original picture

Injected keypoints

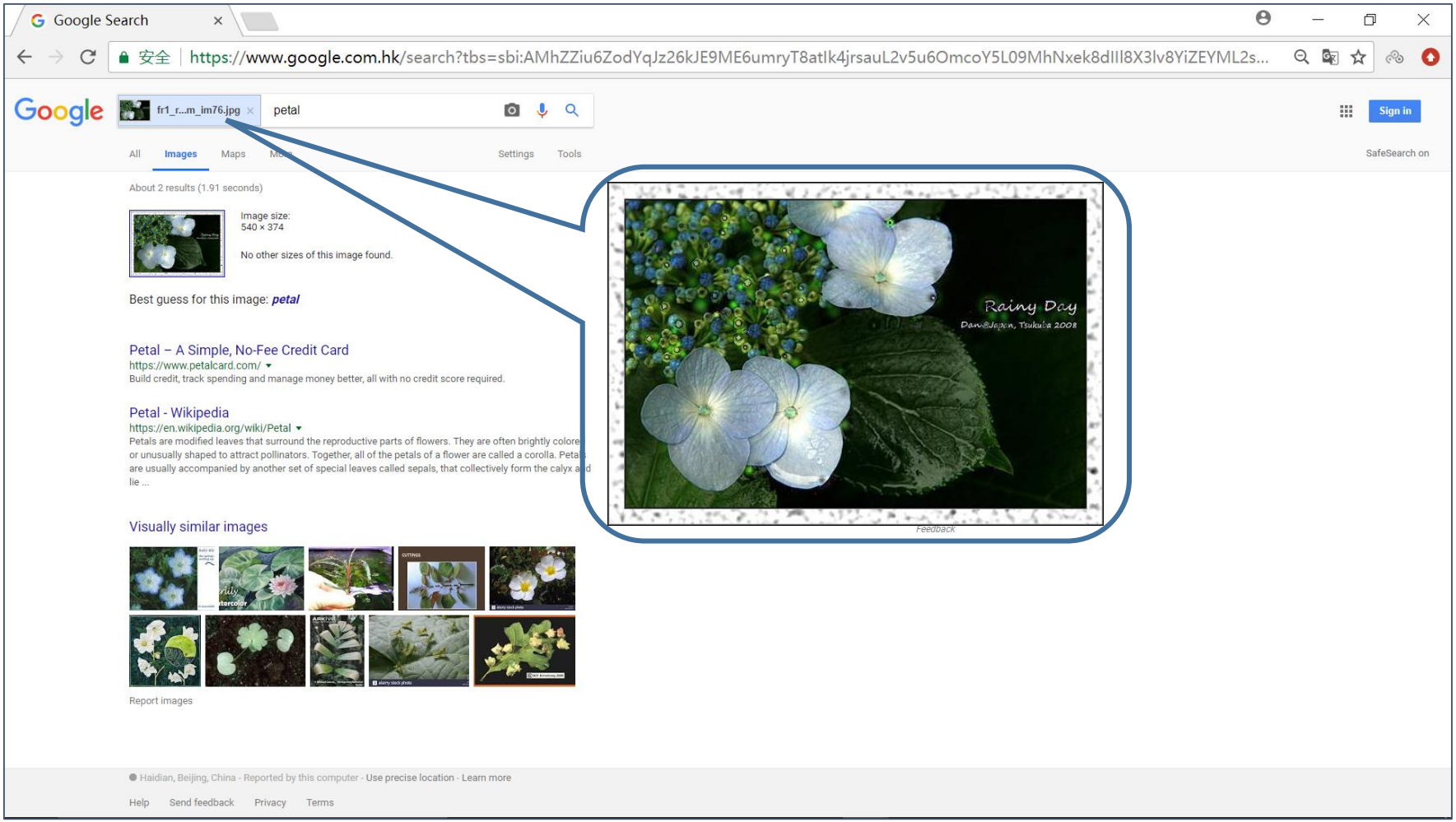
Removed keypoints

New keypoints generated

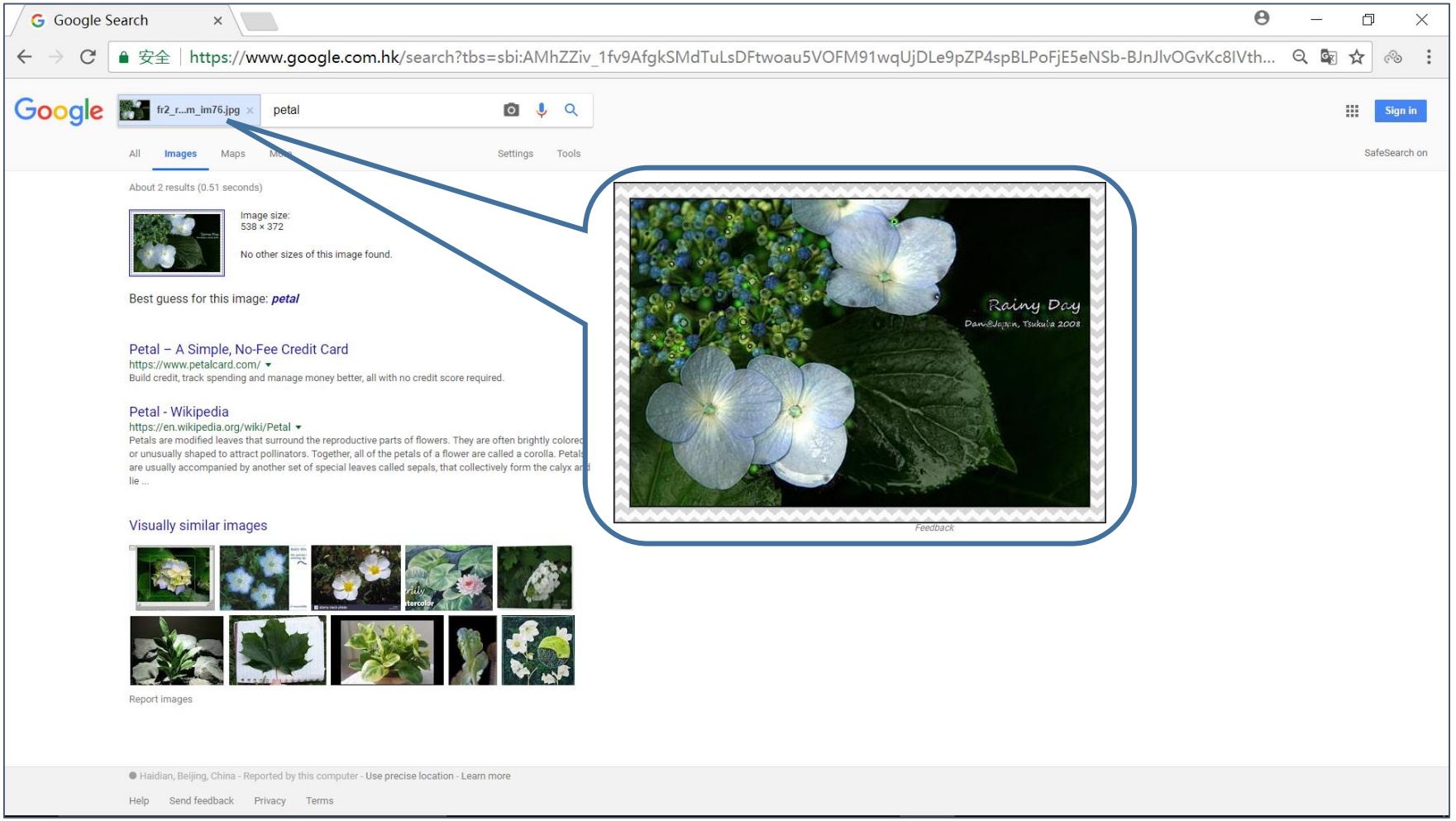





Matching images can be retrieved using original image

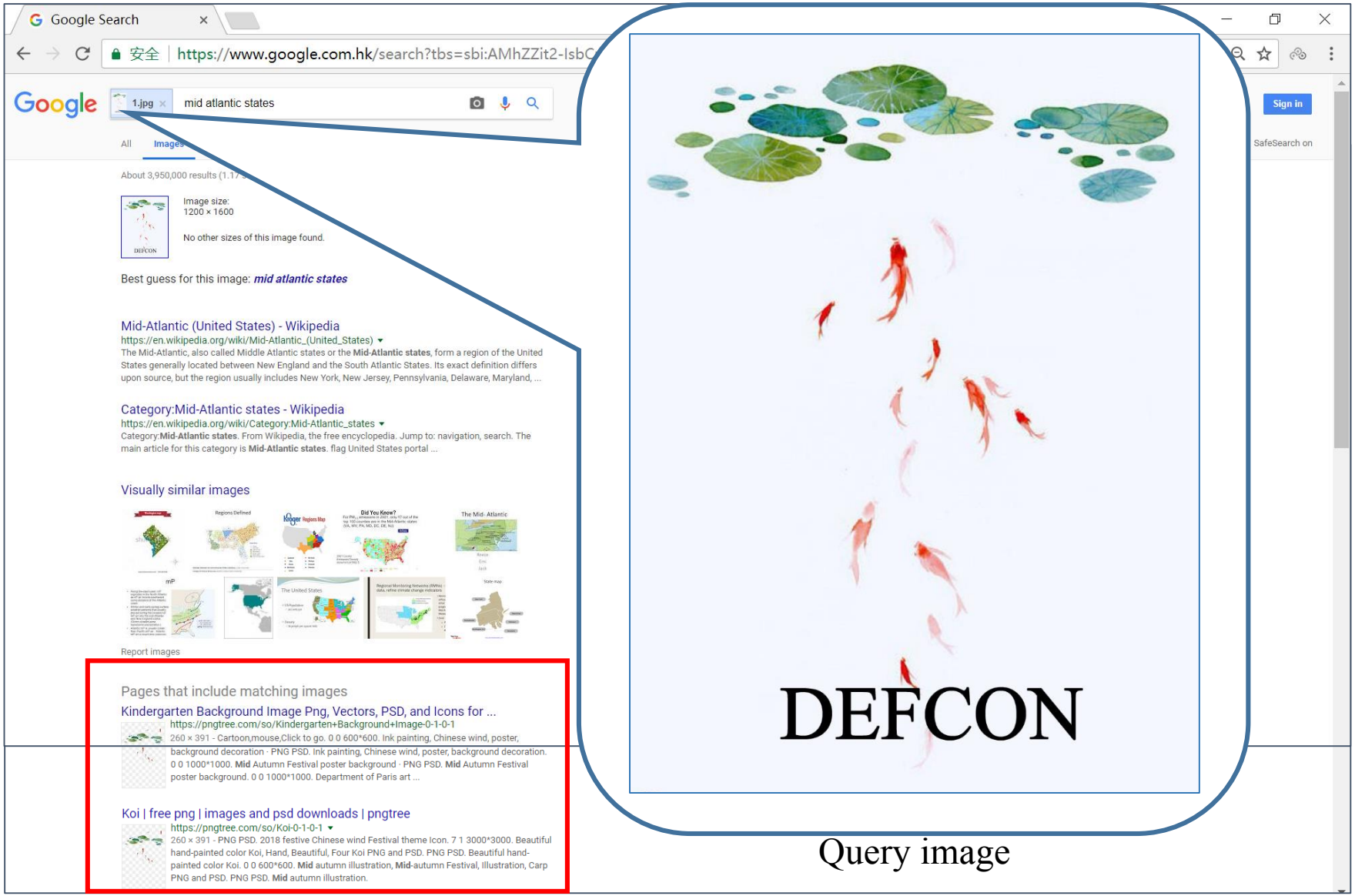



Successfully bypassing Google image search
RMD with **one** iteration + IMD on **15-pixel-wide** frame




Successfully bypassing Google image search RMD with **one** iteration + **basic brick** frame

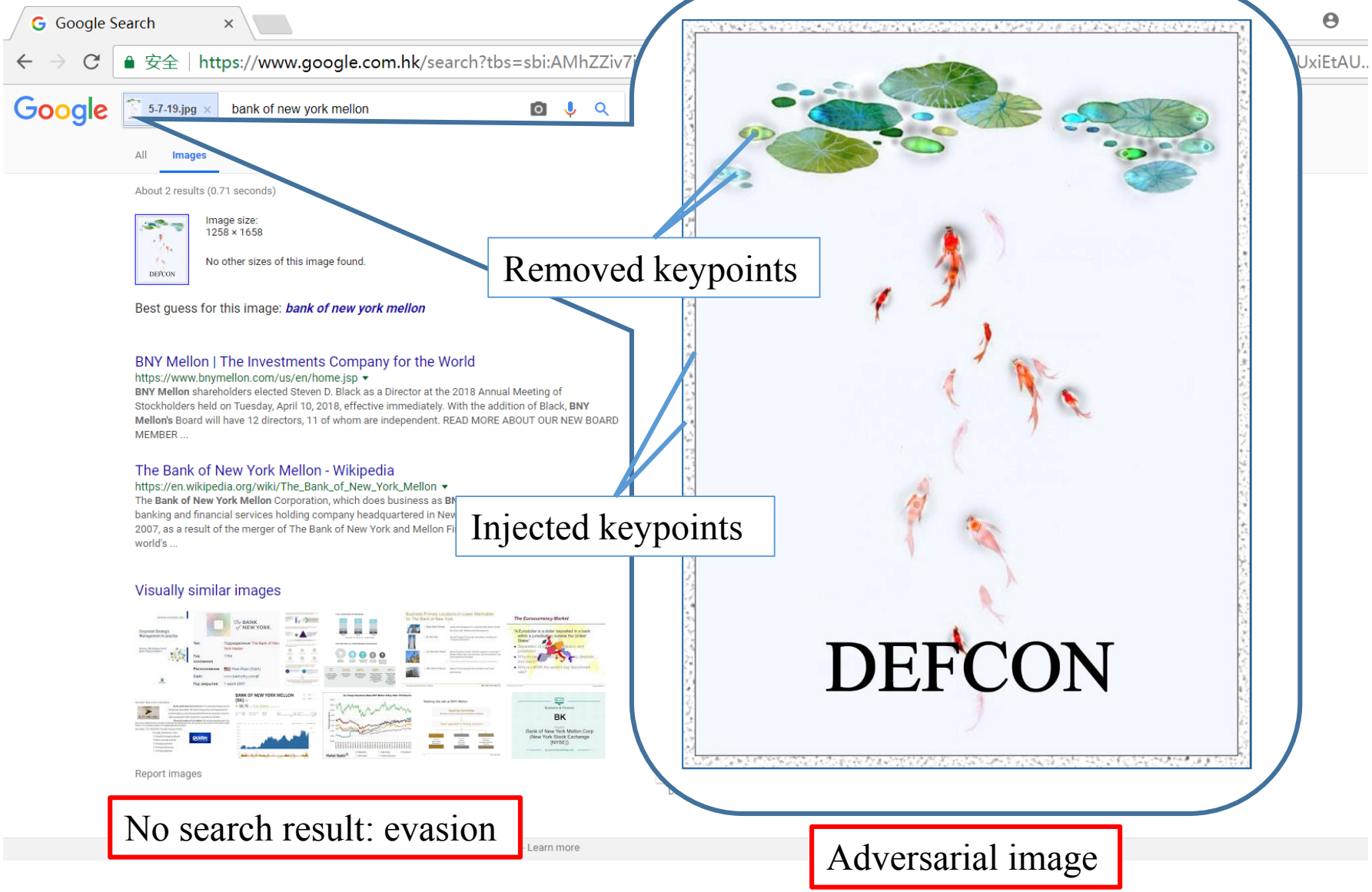
# Evasion




Query image

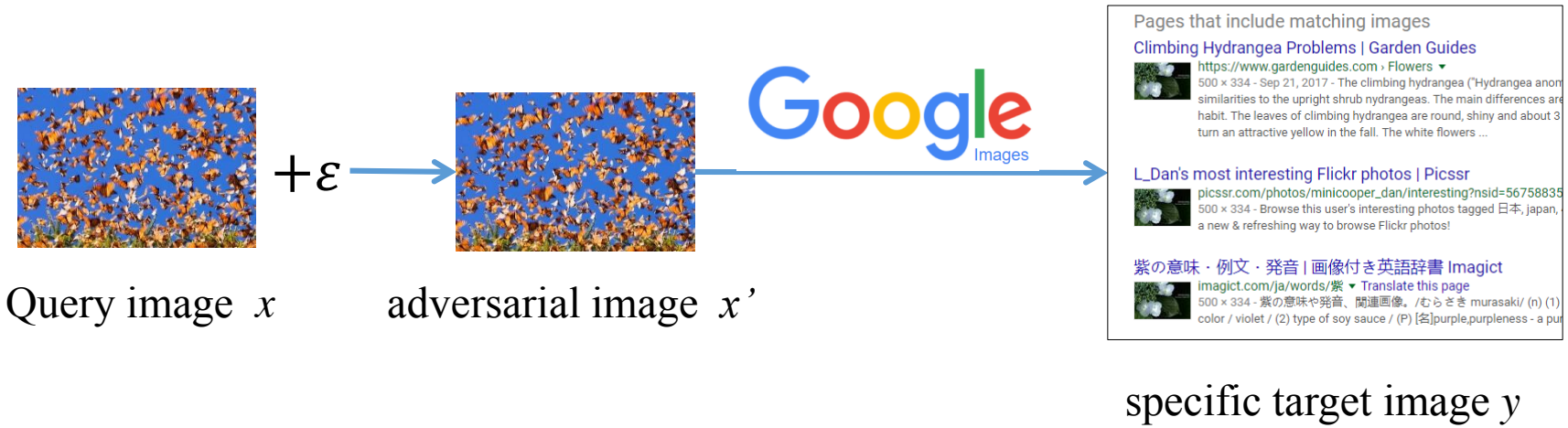
Search result: original image

# Evasion


Google Search
https://www.google.com.hk/search?tbs=sbi:AMhZZiv7
Google
5-7-19.jpg
bank of new york mellon
All
Images
About 2 results (0.71 seconds)
Image size:
1258 × 1658
No other sizes of this image found.
Best guess for this image: bank of new york mellon
BNY Mellon | The Investments Company for the World
https://www.bnymellon.com/us/en/home.jsp
BNY Mellon shareholders elected Steven D. Black as a Director at the 2018 Annual Meeting of Stockholders held on Tuesday, April 10, 2018, effective immediately. With the addition of Black, BNY Mellon's Board will have 12 directors, 11 of whom are independent. READ MORE ABOUT OUR NEW BOARD MEMBER ...
The Bank of New York Mellon - Wikipedia
https://en.wikipedia.org/wiki/The_Bank_of_New_York_Mellon
The Bank of New York Mellon Corporation, which does business as BN
banking and financial services holding company headquartered in New
2007, as a result of the merger of The Bank of New York and Mellon Fi
world's ...
Visually similar images
Report images
Learn more
Removed keypoints
Injected keypoints
DEFCON
No search result: evasion
Adversarial image

# 5. Discussion

- Source/target attack
- A source/target attack means to force the search result of an adversarial image to be a specific target image.

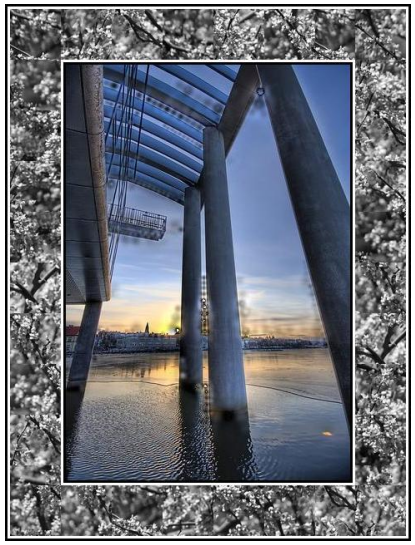



Query image $x$ adversarial image $x'$ specific target image $y$

Threat model

- Source/target attacks will support more sophisticated attacks.

# Source/Target attacking demo



Query image

Query in target system

**VisualIndex**

Injected with keypoints from target image



Search result

- This experiment shows it is possible to achieve source/target attack by injecting keypoints from target image.
- However, severe distortion is caused.
- We will seek for a better strategy in further study.

# 6. Conclusion

- In this work, we present
  - A threat model of evading CBIR systems
  - Several algorithms for removing/injecting keypoints
  - Three bypassing strategies: removal only, injection only and hybrid
  - Evading VisualIndex system
  - Evading Google Image Search Engine
- To conclude, our work proves the existence of threats to CBIR systems and demonstrates that industrial-level image search engines, such as Google Image Search, are prone to be attacked with adversarial images.

# THANKS!

Q&A