

Thumbor ile on-demand resim işleme

Pyİstanbul - 11.07.2015



Murat Çorlu

TrioFan

@muratcorlu

murat.corlu@triofan.com



Resim ne zaman resize edilmeli

- Upload ederken resize et
- Backend'de ilk lazım olduğunda resize et
- Client'ta ilk lazım olduğunda resize et

Upload anında resize

Dezavantajlar

- Görsel katmanı logic ile karıştırmak
- Tasarımda esneklik kaybı
- Varyasyon çıkarma zorluğu

Avantajlar

- Basitlik?

İlk lazım olduğunda resize (SORL)

Dezavantajlar

- Resmin url'i lazım olduğu anda resim de üretiliyor

Avantajlar

- Yönetmesi kolay
- Tasarımda esneklik

İlk lazım olduğunda resize (Client)

Dezavantajlar

- Uzun resim url'leri

Avantajlar

- Yönetmesi kolay
- Tasarımda esneklik
- Resim üretmek gerçekten son gerekli olduğu anda yapılıyor

On-demand resim işleme servisleri

- [Clouinary](#)
- [Imagefly](#)
- [imgix](#)
- [Prism](#)
- ...

The logo for 'thumbor.' is centered within a white rectangular box. This box is set against a dark navy blue background. To the left of the navy blue area is a vertical red stripe. The overall image has a light beige background with faint blue grid lines in the top-left and bottom-right corners.

thumbor.

Neden Thumbor

- Self-hosted, limitsiz
- Open source
- Ücretsiz
- Özellik zengini
- Python ile yazılmış

Nasıl kullanılır?

`http://localhost:8080/unsafe/400/300/smart/filters:
watermark(http://my.site.com/img.png,-10,-10,50)/i.imgur.
com/h2hWy27`

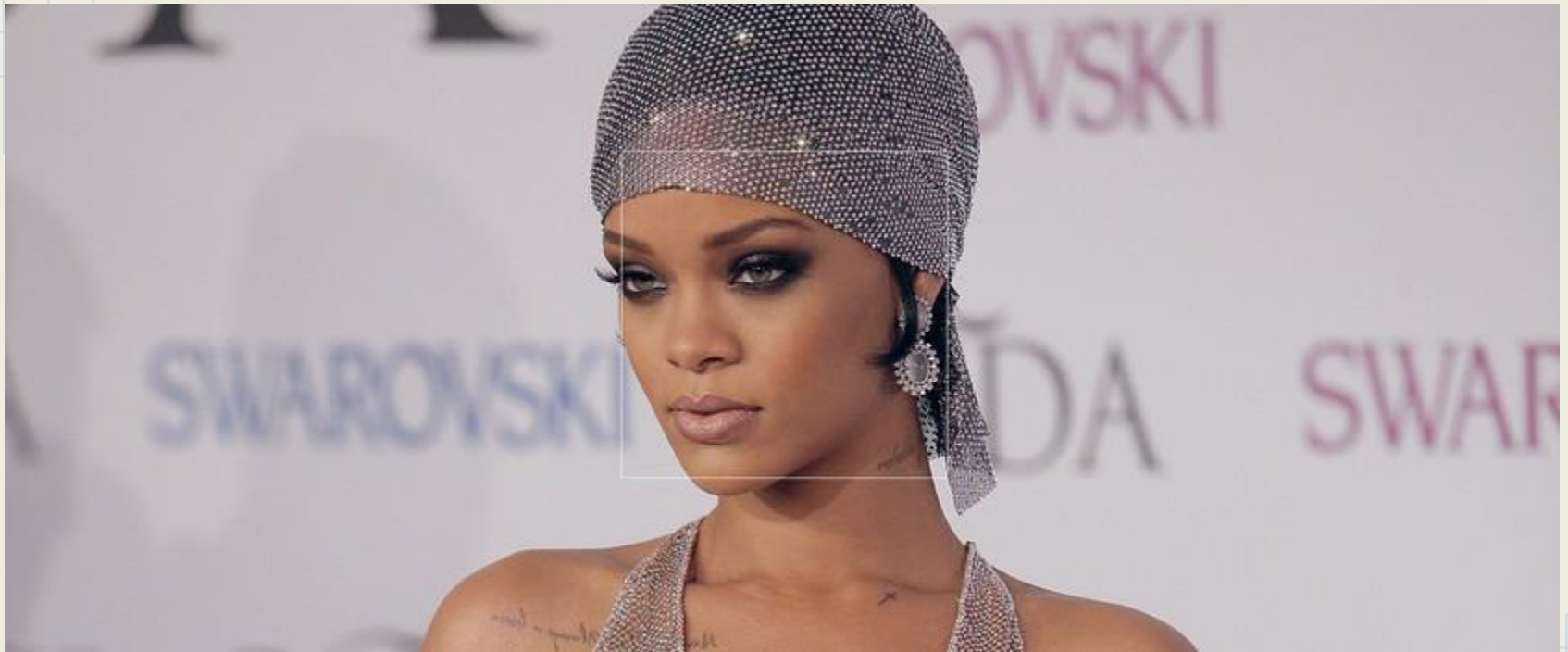
Resim kaynakları (Image Loader)

- HTTP Loader
- File Loader
- MogileFS Loader
- Custom Loaders

Smart Cropping

- OpenCV kullanıyor
- Face detection
- Feature detection

Face Detection



Feature Detection



Original image



The points identified by the good features algorithm

Lazy Smart Cropping

- İlk istekte basit croplama ile cache headersız cevap dönülür
- Arka planda(Redis) resmin smart croplu hali hazırlanır
- Hazırlandıktan sonra cache header'ı ile asıl resim gönderilir.

* Bonus: [RemoteCV](#)

Resim Filtreleri

- Brightness
- Contrast
- Colorize
- Equalize
- Filling
- Format
- Frame
- Extract-Cover
- Grayscale
- Max Bytes
- Noise
- No Upscale
- Quality
- Red-Eye
- RGB
- Round Corners
- Rotate
- Saturation
- Sharpen
- Strip ICC
- Watermark
- Convolution
- Blur
- Extract Focal Points
- GifV
- Curve

Otomatik WebP Desteği

- WebP destekleyen tarayıcılara otomatik olarak WebP formatında cevap dönme
- %20'ye yakın bandwidth tasarrufu

* Dikkat: URL bazlı cache ile kullanılamaz.

Güvenlik?

`http://some.server.com/unsafe/300x301/smart/path/to/image.jpg`

`http://some.server.com/unsafe/300x302/smart/path/to/image.jpg`

`http://some.server.com/unsafe/300x303/smart/path/to/image.jpg`

`...`

`http://some.server.com/unsafe/300x9999/smart/path/to/image.jpg`

`...`

`http://some.server.com/unsafe/9999x9999/smart/path/to/image.jpg`

HMAC Metodu

Kaynak URL:

<http://some.server.com/unsafe/300x200/smart/path/to/image.jpg>

1. HMAC-SHA1 (with SECURITY_KEY)
2. Base64

Sonuç:

<http://some.server.com/1234567890123456789012345678/300x200/smart/path/to/image.jpg>

Django için eklenti?

django-thumbor

```

```

```

```

Upload nasıl olacak?

Thumbor Upload API

- /image REST API
- POST /image : Upload image
- PUT /image : Replace image
- DELETE /image : Delete image

* Authentication desteği yok

Örnek POST isteği:

```
curl -i -H "Content-Type: image/jpeg" -H "Slug: photo.jpg"  
-XPOST http://thumbor-server/image --data-binary  
"@vows/crocodile.jpg"
```

Upload Alternatifi: WebDAV

- Daha generic ve yönetilebilir bir protokol
- Nginx modülü var
- django-resto

Thumbor'u Scale etmek

- Thumbor aynı anda tek bir işlem yapabilir
- İstedığınız kadar instance çalıştırabilirsiniz
- Nginx reverse proxy ile yük dağıtımı yapılabilir

```
http {  
    upstream thumbors {  
        server 127.0.0.1:8001;  
        server 127.0.0.1:8002;  
        server 127.0.0.1:8003;  
    }  
  
    server {  
        listen 80;  
  
        location / {  
            proxy_pass http://thumbors;  
        }  
    }  
}
```




Soru?



Teşekkürler

@muratcorlu