



CDNetworks

Accelerate. Secure. Control.

Image Processing Value Added Service

Market Version 2.0

October 22, 2024

Contents

1 VAS Intro	3
1.1 Brief Introduction.....	3
1.2 Applicable Product Lines.....	3
1.3 Application Scenarios.....	3
2 Feature Detail.....	4
2.1 Image Compression	4
2.2 Image Cropping.....	5
2.3 Image Scaling.....	14
2.4 Image Rotating.....	21
2.5 Image Adaptive Rotation.....	21
2.6 Image Format Conversion.....	22
2.7 Image Gray Scaling.....	23
2.8 Image Watermarking.....	23
2.9 Image Information Obtaining	24
2.10 Image Brightness and Contrast.....	27
2.11 Fuzzy Processing.....	27
2.12 Background Color.....	27
2.13 Interpolation	31
3 Key Benefits.....	31

1 VAS Intro

1.1 Brief Introduction

Integrated to CDN architecture, image processing service provides a one-stop image processing solution that includes format conversion, cropping, scaling, rotating, watermarking, etc. This service saves customers from image processing and reduces the ratio of back-to-origin requests. Besides, by outputting the adaptive content with high-reliability via the proximate CDN network, the speed of pages visiting is increased.

1.2 Applicable Product Lines

- Content Acceleration
- Dynamic Web Acceleration

1.3 Application Scenarios

1) Websites with high-volume image processing needs, particularly in e-commerce, news portals, social media, and user-generated content (UGC) platforms.

2) Websites focused on optimizing image processing costs, performance, stability, and ease of use.

3) Mobile networks and devices, where image processing significantly enhances mobile acceleration and performance.

4) Global businesses seeking localized image processing to meet the specific requirements of different markets and enhance website traffic in various regions

Note: Some image-related services can be customized by URL rewriting. For example, if a customer was previously using a different CDN's image optimization features (which likely have different URL parameters for resizing, cropping, etc.), CDNetworks can rewrite those incoming URLs to match its own system. This allows customers to switch to CDNetworks without having to change the image URLs embedded in their website's code. Essentially, it's a compatibility feature to simplify migration and minimize disruption. Please contact CDNetworks technical support team for details.

2 VAS Detail

Image processing service includes compression, cropping, scaling, rotation, format conversion, gray scaling, watermarking, image information obtaining, etc. It supports simultaneous enabling of most operation parameters corresponding to different functions. And those parameters that do not support concurrent operations, such as file information obtaining parameters, will take effect by separate steps.

The image processing is made by an image processing server based on the parameters from the image request url. The formats of parameters from the url are fixed, which can be divided into two levels: For level-one parameters, "&" is the separator, "=" assigns the value, and the keywords are {q, Q, crop, resize, r, o, f, g, watermark, info}. For level-two parameters, "," is the separator, "_" assigns the value, and the keywords are {p, w, h, x, y, a, g, s, e, t, r, s, image, text, d, front, color, size, bc, bs}. When a request is configured for image processing, these keywords in parameters are regarded as the image processing parameters. With the combination of multiple parameters, complex image processing can be realized to meet various user requirements of different scenarios.

2.1 Image Compression

1) Overview

Image quality compression. The lower the parameter of compression quality is, the higher the compression ratio and the smaller the Content-length will be.

Absolute quality	Compress the original image with q. If the original image quality is 90% and q=80, then a new image at 80% quality is generated.
Relative quality	Compress the original image with Q. If the original image quality is 90% and Q=80, then a new image at 72% quality is generated.

2) Parameters

Parameters: q, Q

Value range: 1-100

Default value: 0, no compression

3) Example

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?q=80>

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?Q=80>

4) Notices

- When both q and Q exist, q is effective.
- If the quality of the original image is lower than q, processing will not take effect, and the original image will be returned.
- Suggested value: 75, as the value being too small may lead to distortion. Please be noted that the percentages used above are only for explanation, not the actual quality value of the image.
- Only jpg and webp images formats are supported. And webp images with the parameter Q are processed by the q method by default.

2.2 Image Cropping

1) Overview

There are four types of image cropping: anchor point cropping, center cropping, gravity center cropping and index cropping. Besides, you can specify the cropping methods, such as square cropping, empty areas filling after cropping, etc.

Level-one parameter	Level-two parameter	Parameter value	Operation description	Value range
crop=	p_	0	Anchor point cropping Set the anchor coordinate (x, y) in the top left corner of the original image, and crop the image with the specified width and height in pixel (w,h). When the specified height (or width) is equal to 0 or	w,h value range:[0, +∞)

			<p>the width (or height) counting from the starting point exceeds the original image, it is directly cropped to the end of the original image. If the starting points exceed the original image, return the original image.</p> <p>Example: start from coordinate x_50, y_50, crop a 400*200 image http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_0,x_50,y_50,w_400,h_200</p>	
		1	<p>Center cropping</p> <p>Take the center of the original image as the image center, and crop out an image with the specified width and height in pixel (w,h). When either w or h is invalid, return the original image.</p> <p>Example: take the original image center as the cropping center, and crop out a 400*200 image http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_1,w_400,h_200</p> <p>Note: corresponding old parameters, p=4</p>	w,h value≥0
		2	<p>Gravity center cropping</p> <p>Use parameter g to set the position of nine-rectangle-grid, and crop an image with specified width and height in pixel</p>	<p>x,y value≥0, the default is 0; w,h value≥0; g value</p>

			<p>(w,h) after offsetting x and y. When either w or h is invalid, return the original image. Example: start from southeast, crop a 400*200 image http://images.demo.cdnetworks.com/ctuU-fxriqxx2975432.jpg?crop=p_2,w_400,h_200,g_se</p>	<p>range: [nw,north,ne,west,center,east,sw,south,se].</p>
		3	<p>Index cropping Crop the x axis (y axis) of the original image at the specified length, and then get the image within the area with specific index. The length range is [1, length of the cropped side]. If the input length exceeds the cropped side, return the original image. The principle also applies in index cropping when the input index exceeds the maximum. Example: crop y axis with the length of 100, and crop out the image within the corresponding area with specified index 2 http://images.demo.cdnetworks.com/ctuU-fxriqxx2975432.jpg?crop=p_2,i_2,y_100</p>	<p>x,y value\geq0, default is 0; w,h value\geq0; i value\geq0, default is 0.</p>
		4	<p>Inscribed circle Show the original image in a circle shape If the final image format is png, webp or other formats that support transparency (also</p>	<p>r value\geq0</p>

			<p>known as Alpha channel), then fill the non-circle area with transparency. If the final image format is jpg, fill the non-circle area with white.</p> <p>Value range: r value is larger than 0 ; inscribed circle and scaling can be effective at the same time.</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_4,r_200</p>	
		6	<p>square_enlarge cropping</p> <p>When the parameter s is 1, crop a square with the shorter side of the original image being the square side.</p> <hr/> <p>When the parameters e and s are 1, enlarge the square using the longer side of the original image.</p> <p>For example: enlarge the square with the longer side of the original image being the square side.</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_6,s_1,e_1</p>	<p>s value range is 0-1, no default value;</p> <p>e value range is 0-1, no default value.</p>
		7	<p>extend cropping (value of t is 1)</p> <p>When the requested width and height are greater than that of the original image, center the original image and fill the surrounding</p>	

			<p>with white;</p> <p>When the requested width and height are smaller than that of the original image, center crop the image;</p> <p>When the requested width is larger than that of the original image, and height is smaller, center crop the image and fill both left and right sides with white; When the requested width is smaller than that of the original image, and height is larger, center crop the image and fill both top and bottom sides with white;</p> <p>For example: the size of original image is 640x354, and the requested size is 700x500, center crop the original image and fill the surrounding with white.</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_7,w_700,h_500</p>	
--	--	--	---	--

		9	<p>Percentage cropping</p> <p>w, h: indicates that the width and height of thumbnail are w%W and h%H, and the value range is [1, 1000];</p> <p>When w and h are specified, the image with the size of width and height in pixel (w%W, h%H) will be cropped from the start point of the original image;</p> <p>When w is specified, the image with the size of width and height in pixel (w%W, w%H) will be cropped from the start point of the original image;</p> <p>When h is specified, the image with the size of width and height in pixel (h%W, h%H) will be cropped from the start point of the original image.</p> <p>x,y: are the start anchor coordinates;</p> <p>When x,y are specified, the anchor coordinates are (x,y);</p> <p>When x is specified, anchor coordinate is (x, (H-h)/2);</p> <p>When y is specified, anchor coordinate is ((W-w)/2, y);</p> <p>Note: w and h are the width and height in pixel after percentage calculation;</p> <p>For example: crop an image with 50% width and 70% height of the</p>	<p>w,h value range is [1, 1000];</p> <p>x,y value range is [0, +∞).</p>
--	--	---	---	---

			<p>original image.</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_9,w_50,h_70</p>	
		10	<p>Gravity center cropping after scaling</p> <p>Scale down the original image, and the width and height of the re-sized image are larger or equal to $w*h$; Crop by specified g, and a thumbnail image of $w*h$ will be obtained;</p> <p>When w or h is larger than that of the original image, the corresponding value of the original image is adopted;</p> <p>When w or h equals to 0, make the proportional scaling with the</p>	<p>w,h cannot both be 0;</p> <p>w,h value $[0, +\infty)$</p>

			other side; For example: resize the original image with the proportion of 300x70, and adopt the upper part of gravity center	
	x,y	x-coordinate value, y-coordinate value		
	w,h	Width, height value		
	g	[nw,north, ne,west,center,east,sw,south,se]	Use parameter g to set the position of the nine-rectangle-grid	Default is se
	i	Specify index block	After cropping into designated length, specify the index	Default is 0
	s,e	0 or 1	Crop by a specified method	
	t	0 or 1	Crop by a specified method	

2) Example

Anchor point cropping

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_0,x_50,y_50,w_400,h_200

or

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=x_50,y_50,w_400,h_200

Center cropping

http://images.example.com/ctuU-fxriqqx2975432.jpg?crop=p_1,w_400,h_200

Gravity center cropping

http://images.example.com/ctuU-fxriqqx2975432.jpg?crop=p_2,w_400,h_200,g_se

Index cropping

http://images.example.com/ctuU-fxriqqx2975432.jpg?crop=p_3,i_2,y_100

Inscribed circle

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_4,r_200

square_enlarge cropping

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_6,s_1,e_1

extend cropping

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_7,w_700,h_500

Percentage cropping

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_9,w_50,h_70

Gravity center cropping after scaling

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?crop=p_10,w_300,h_70,g_north

3) Notices

- Please choose the cropping method and cropping size based on the image resolution and avoid using the unreasonable parameters.
- When the specified width (or height) from the start point exceeds that of the original image, crop the entire image of the original image.
- To avoid potential conflicts with newly-added parameters, processing parameters of the first version image shall be fully replaced with the new ones.

2.3 Image Scaling/Resizing

1) Overview

Image scaling is changing the width and height pixel (w for width and h for height). See below for some common scaling methods:

Level-one parameter	Level-two parameter	Parameter value	Operation description	Value range
resize=	p_	0	Compulsory stretch Stretch the original image following w*h pixel. If there is no w*h value, use that of the original image. Parameters of w and h must coexist, otherwise the original image will be returned Example: stretch the original image, and the width and height in pixel is 500*200 http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_0,w_500,h_200 Note: corresponding old parameter p=0	w,h value range is [1-10000]
		1	Percentage scaling Resize the width and height of the original image by a percentage For example: scale down the width to 50% of the original, and height to 50% of the original http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_1,w_50,h_50 ; Note: corresponding old parameter p=5	w,h value range is [1-1000]
		2	Filling after scaling	w,h value

			<p>Set w and h, scale the original image proportionally. The width and height should not surpass w*h. Center the resized image and fill the empty part with color. When either w or h is invalid, return the original image.</p> <p>Example: scale down the original image to 200*200 and fill in the empty space with white</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_2,w_200,h_200;</p>	<p>range is [1-10000];</p> <p>Color supports hexadecimal, and the default is #ffffff(white)</p>
		3	<p>Cropping after scaling</p> <p>1) Specify w and h: scale down proportionally, and then crop the part that exceeds w*h, and the processed image is w*h.</p> <p>2) Only specify w: if w is smaller than width and height, then scale down first before cropping, the processed image is w*w; If w is smaller than width while greater than height, then crop the image, and the processed image is w*height; If w is greater than width but smaller than height, then crop the image, and the processed image is width*w; If w is greater than width and height, the original image will be returned</p> <p>3) Only specify h: if h is smaller than width and height at the same time, then scale down first before cropping, and</p>	<p>w,h value range is [1-10000]</p>

		<p>the processed size is $h \times h$; if h is smaller than width while greater than height, then crop the image, the processed size is $h \times \text{height}$; if h is greater than width but smaller than height, then crop the image, and the processed size is $\text{width} \times h$; if h is greater than width and height, return the original image</p> <p>Example: scale down the original image, and the size of the processed image is 300×200</p> <p>http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_3,w_300,h_200;</p> <p>Note: Corresponding old parameter $p=1$</p>	
	4	<p>Scale down proportionally, and the width and height should not surpass $w \times h$</p> <p>1) Specify w and h: width is w, scale down height proportionally; height is h, scale down the width proportionally. It is correct when the width and height do not surpass $w \times h$</p> <p>2) Only specify w: w for width, scale down height proportionally</p> <p>3) Only specify h: h for height, scale down width proportionally</p> <p>Note: corresponding old parameter $p=2$</p>	<p>w, h value range is [1-10000]</p>
	5	<p>Scale down proportionally, and width and height should be no smaller than $w \times h$</p> <p>1) Specify w and h: width is w, scale down width proportionally; height is h,</p>	<p>w, h value range is [1-10000]</p>

			<p>scale down the height proportionally, and the correct result is that width and height are no smaller than $w \cdot h$</p> <p>2) Only specify w: when width is w, scale down height proportionally</p> <p>3) Only specify h: when height is h, scale down width proportionally</p> <p>Note: corresponding old parameter $p=3$</p>	
		6	<p>Scale up proportionally, and if both w and h are smaller than that of the original image, width and height cannot surpass $w \cdot h$</p> <p>1) If width of the original image is smaller than the given w, and height of the original image is greater than the specified h, return the original image</p> <p>2) If width of the original image is greater than the given w, and height of the original image is smaller than the specified h, return the original image</p> <p>3) If the width of the original image is greater than the given w, and the height of the original image is greater than the specified h, scale the image according to the given w, h proportionally. Only adopt the values that do not exceed w, h.</p> <p>Note: corresponding old parameter is $p=8$</p>	<p>w, h value range is [1-10000]</p>

		7	<p>Proportional scaling, and the pixel should not exceed the specified value</p> <p>Make the proportional scaling of the original image, and the value of width*height shall not exceed the pixel value of the specified area.</p> <p>Example: specify 10000 as the pixel value, and make the proportional scaling of the image</p> <p>http://images.demo.cdnetworks.com/ctuU-</p>	<p>a value range is [1-24999999]</p>
--	--	---	--	--------------------------------------

		8	<p>Width and height limit</p> <p>When the requested image is larger than a certain width and height, make the proportional scaling; the width and height of thumbnail image should be smaller or equal to the specified; or there will be no processing;</p> <p>When w and h are specified, and the width and height of the original are smaller than the specified value, there will be no processing; otherwise, there will be proportional scaling down, and the width and height of thumbnail are smaller or equal to w*h;</p> <p>When w is specified, and the width of the original is smaller than the specified value, there will be no processing; otherwise, there will be proportional scaling down with according to w;</p> <p>When h is specified, and the height of the original is smaller than the specified value, there will be no processing; otherwise, there will be proportional</p>	<p>w,h value range is [1-10000]</p>
	w,h	Width , height	When value range of p is 0,2,3,4,5, w and h specify the width and height of the thumbnail	
	color	Color code	For color code, only hexadecimal value starts with # is supported	Default is #ffffff(white)
	a	Width *height	When the value range of p is 7, scale the image proportionally and pixel of width*height should not exceed the a-specified pixel	A value range is [1-24999999]

2) Example

Compulsory stretching

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_0,w_500,h_200

Percentage scaling

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_1,w_50,h_50

Filling after scaling

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_2,w_200,h_200

Cropping after scaling

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_3,w_300,h_200

Scaling down, and the height and width should not exceed $w*h$

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_4,w_300

Scaling down, and the height and width should not be smaller than $w*h$

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_5,w_300,h_200

Proportional scaling, and the pixel should not exceed the specified value

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_7,a_10000

3) Notice

- Usually image scaling down instead of scaling up is recommended in this scenario, as the oversized original image may prolong response time.
- Usually proportional scaling is more frequently adopted; width and height should not surpass $w*h$, and `p_4` needs to be specified.
- For filling after scaling with corresponding parameters:

resize=p_2,w_<width_value>,h_<hight_value>,color_<color_value>, currently it will only support jpg/jpeg format;If the width(w) or height(h) value is invalid, then the original image is returned.

2.4 Image Rotating

1) Overview

Rotate the image clockwise, and set the white background for rotated image.

2) Parameters

Parameter: r

Value range: 0-360

Default value: 0

3) Example

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?r=30>

2.5 Image Adaptive Rotation

1) Overview

Photos taken by some mobile phones may contain rotation parameters (stored in the exif information of the photo), so configuration can be set to determine whether to make adaptive rotation of the photos.

O=0, means following the default rotation orientation of the original image instead of auto rotation;

O=1, means auto rotating the image following the rotation parameters of the image, and if there exist the scale-down parameters, rotate the image after scaling down the image;

O=2, means auto rotating the image following the rotation parameters of the image, and if there exist the rotation parameters, rotate the image before scaling down the image.

2) Parameters

Parameter: o

Value range: [0,1,2]

Default value: 0

3) Example

Scale the image proportionally, then conduct adaptive rotation

http://images.demo.cdnetworks.com/orientation1.jpg?resize=p_5,w_300&o=1

Make adaptive rotation first, then scale the image proportionally

http://images.demo.cdnetworks.com/orientation1.jpg?resize=p_5,w_300&o=2

4) Notice

Function can be realized by adopting parameter o and auto_orient that is configured in image processing, but parameter o has higher priority.

2.6 Image Format Conversion

1) Overview

Format conversion among four image formats (jpeg/jpg, png, gif, webp) is supported. After conversion, the respond Content-Type is the corresponding image type.

2) Parameters

Parameter: f

Value range: jpg, png, gif, webp, AVIF (developing, test environment available)

Default value: jpg

3) Example

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?f=webp>

2.7 Image Gray Scaling

1) Overview

Remove image colors and leave only gray scale.

2) Parameters

Parameter: g

Value range: 0, 1

Default value: 0

3) Example

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?g=1>

2.8 Image Watermarking

1) Overview

Supports image watermarking, text watermarking and mixed watermarking.

Level-one parameter	Level-two parameter	Parameters and operation description
watermark=	m_	Modes of watermark processing 0: no watermarking processing, default value; 1. Image watermarking; 2. Text watermarking; 3. Mixed watermarking, image under text; 4. Mixed watermarking, text under image;
	image_	Base64 encoding, support pre-processing of watermark image
	text_	Base64 encoding
	d_	Watermark transparency, value range: 0-100
	g_	Watermark position, value range: [nw,north,ne,west,center,east,sw,south,se]
	dx_	Offset on X-axis, value range is $[1,+\infty)$, default value is 10, unit:

		pixel
	dy_	Offset on Y-axis, value range is $[1, +\infty)$, default value is 10, unit: pixel
	font_	Font of text watermark, base64 encoding
	color_	Color of watermark text, with white as the default value. More options: "blue", "#0000ff", "rgb (0,0,255)", base 64 encoding
	size_	Font size of text watermark, value range is (0,1000], default value is 30
	r_	Rotation of text watermark, value range is [0,360], default value is 0
	bc_	Outlined color of the watermark text, base64 encoding
	bs_	Outlined thickness of the watermark text, range value is (0, 50), default value is 1, suggested value is 0.5-2
	percentt_	controls the scaling of the watermark image's shorter edge as a percentage, with a valid range of 0 to 100 (integers).

2) Example

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?watermark=m_1,image_aHR0cDovL2ltYWdlLWRibW8uaW1nLWNuLWWhhbmd6aG91LmFsaXI1bmNzLmNvbS9wYW5kYS5wbmc/cmVzaXplPXBfMSx3XzUwLGhfNTA=

3) Notices

- Parameters of image processing should be within the value range, and if the value is not specified, adopt the default values assigned.
- Pre-processing of watermark image support percentage scaling.
- No default value for bordercolor, and only when this field is set with a specific value, will the outlining take effect.

2.9 Image Information Obtaining

1) Overview

Obtain the basic or exif information of the image.

2) Parameters

Parameter: info

Value range: [0,1,2], 0 means no information obtaining, 1 means basic info obtaining, and 2 means exif info obtaining.

Default value: 0

3) Example

Get the basic information of the image:

<http://images.demo.cdnetworks.com//ctuU-fxriqqx2975432.jpg?info=1>

```
{
  "FileSize": {"value": "175839"},
  "Format": {"value": "jpeg"},
  "ImageHeight": {"value": "354"},
  "ImageWidth": {"value": "5184"},
  "XResolution": {"value": "72"},
  "YResolution": {"value": "72"},
  "ImageLength": {"value": "3456"},
  "BitsPerSample": {"value": "8, 8, 8"},
  "PhotometricInterpretation": {"value": "RGB"},
  "ImageDescription": {"value": "attends the 2016 Laureus World Sports Awards at
Messe Berlin on April 18, 2016 in Berlin, Germany."},
  "Make": {"value": "Canon"},
  "Model": {"value": "Canon EOS-1D X"},
  "SamplesPerPixel": {"value": "3"},
  "ResolutionUnit": {"value": "Inch"},
  "Software": {"value": "www.meitu.com"},
  "DateTime": {"value": "2016:04:18 19:26:05"},
  "Artist": {"value": "Ian Walton"},
  "YCbCrPositioning": {"value": "Co-sited"},
  "Copyright": {"value": "2016 Getty Images (Photographer) - (Editor)"},
```

```
"ExposureTime": {"value": "1/250 sec."},
"FNumber": {"value": "f/7.1"},
"ExposureProgram": {"value": "Manual"},
"ISOSpeedRatings": {"value": "400"},
"ExifVersion": {"value": "Unknown Exif Version"},
"DateTimeOriginal": {"value": "2016:04:18 18:13:51"},
"DateTimeDigitized": {"value": "2016:04:18 18:13:51"},
"ComponentsConfiguration": {"value": "Y Cb Cr -"},
"ShutterSpeedValue": {"value": "8.00 EV (1/256 sec.)"},
"ApertureValue": {"value": "5.62 EV (f/7.0)"},
"ExposureBiasValue": {"value": "0.00 EV"},
"MaxApertureValue": {"value": "3.00 EV (f/2.8)"},
"MeteringMode": {"value": "Pattern"},
"Flash": {"value": "Flash did not fire, compulsory flash mode"},
"FocalLength": {"value": "50.0 mm"},
"UserComment": {"value": ""},
"SubsecTime": {"value": "68"},
"SubSecTimeOriginal": {"value": "68"},
"SubSecTimeDigitized": {"value": "68"},
"FlashPixVersion": {"value": "FlashPix Version 1.0"},
"ColorSpace": {"value": "sRGB"},
"PixelXDimension": {"value": "5184"},
"PixelYDimension": {"value": "3456"},
"FocalPlaneXResolution": {"value": "3545.8276"},
"FocalPlaneYResolution": {"value": "3526.531"},
"FocalPlaneResolutionUnit": {"value": "Inch"},
"CustomRendered": {"value": "Normal process"},
"ExposureMode": {"value": "Manual exposure"},
"WhiteBalance": {"value": "Manual white balance"},
"SceneCaptureType": {"value": "Standard"},
"GPSVersionID": {"value": "2.3.0.0"}}
```

If there is no EXIF information, respond message: no exif data.

2.10 Image Brightness and Contrast

1) Overview

Adjust image brightness and contrast.

2) Parameters

Parameters: bright, contrast

Value range: -100-100

Default value: 0

3) Example:

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?bright=10>

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?contrast=-10>

2.11 Fuzzy Processing

1) Overview

Blur the image

2) Parameter: blur

Level-two parameter	Description	Value
r	Radius of Gaussian blur	Value range is 1-50
s	standard deviation of the Gaussian distribution	Value range is 1-50

(3) Example:

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?blur=r_2,s_4

2.12 Background Color

1) Overview

Set the background color of the image.

2) Parameter: bgcolor=

Value range: English color words, ~hexadecimal

3) Example:

<http://images.demo.cdnetworks.com/ctuU-fxriqqx295432.png?bgcolor=~dddfff>

2.13 sharpness

1) Overview

This feature allows you to adjust the sharpness of delivered images.

2) Parameter

Parameter: sharpen

Usage: Append sharpen=<value> to the image URL.

Value Range: 0-100 (integer). 0 represents no sharpening, 100 represents maximum sharpening.

Effect: Higher values result in sharper images. However, excessive sharpening can introduce undesirable artifacts. Use with caution.

Example

To apply a sharpening level of 1 to the image image.jpg, use the following URL:

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?sharpen=1>

4) Notice

Only support static image.

2.14 Progressive Image Loading

1) Overview

This feature enhances perceived image loading speed by using interlacing. Interlacing displays a low-resolution version of the image first, then progressively refines the quality until the full-resolution image is shown.

2) Parameter

Parameter: interlace

Usage: Append interlace=<value> to the image URL.

Values:

0: Standard progressive display (no interlacing).

1: Interlaced display.

3) Example

<http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?interlace=1>

4) Notices

Supported Formats: JPEG/JPG

2.15 GIF Frame Extraction

1) Overview

This feature allows you to extract individual frames from an animated GIF.

(2) Parameter

Parameter: extract=s_e (or extract=s_)

Usage: Append the extract parameter to the GIF URL.

Level-two Parameter	Description	Value
s	Start Frame	Integer [0, +∞)

e	End Frame	Integer [0, +∞)
---	-----------	-----------------

(3) Examples

Extract the first frame:

http://images.demo.cdnetworks.com/pink.gif?extract=s_1,e_1

Extract frames 2 through 20:

http://images.demo.cdnetworks.com/pink.gif?extract=s_2,e_20

Extract from frame 0 to the last frame::

http://images.demo.cdnetworks.com/pink.gif?extract=s_0,e_0

(4) Notices

This feature only supports GIF images.

2.16 Image Border

1) Overview

This feature adds a border to an image.

2) Parameter

Parameter: border=w_<width>,color_<color>

Usage: Append the border parameter to the image URL.

Note: Both the w and color sub-parameters must be present. If you omit w, the default width of 1 pixel will be used. However, omitting color will disable the border entirely.

Level-two parameter	Description	Valid Values
w	Border width in pixels	[1, +∞), integer, defaults to 1
color	Border color	Hexadecimal color code (e.g., red, #FF0000,

		rgb(255,0,0)
--	--	--------------

(3) Example:

http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?border=w_15,color_red

(4) Notices

Supported Formats: JPEG, PNG, WebP

2.17 Interpolation

1) Overview

Set the interpolation algorithm of image.

2) Parameter: interpolation=

Value range: bilinear, bicubic, nearest-neighbor, spline

Note: interpolation algorithm will only be effective when using with scaling.

3) Example: http://images.demo.cdnetworks.com/ctuU-fxriqqx2975432.jpg?resize=p_0,w_100,h_100&interpolation=bilinear

3 Key Benefits

- 1) Reduce storage and image optimization costs on origin sites. Customers can store only one single image, while CDNetworks' image processing feature helps generate various images for different needs. This greatly offloads the image transformation of applications and servers at customer origin.
- 2) End-users can enjoy suitable images faster from CDN edge PoPs, without any quality compromise.