<?php

/\*\*

\*

\* 鐩镐技鍥剧墖鎼滅储hash鐨刾hp瀹炵幇

\* @author welefen

\*

\*/

class Imghash{

private static $\_instance = null;

public $rate = 2;

public static function getInstance(){

if (self::$\_instance === null){

self::$\_instance = new self();

}

return self::$\_instance;

}

public function run($file){

if (!function\_exists('imagecreatetruecolor')){

throw new Exception('must load gd lib', 1);

}

$isString = false;

if (is\_string($file)){

$file = array($file);

$isString = true;

}

$result = array();

foreach ($file as $f){

$result[] = $this->hash($f);

}

return $isString ? $result[0] : $result;

}

public function checkIsSimilarImg($imgHash, $otherImgHash){

if (file\_exists($imgHash) && file\_exists($otherImgHash)){

$imgHash = $this->run($imgHash);

$otherImgHash = $this->run($otherImgHash);

}

if (strlen($imgHash) !== strlen($otherImgHash)) return false;

$count = 0;

$len = strlen($imgHash);

for($i=0;$i<$len;$i++){

if ($imgHash{$i} !== $otherImgHash{$i}){

$count++;

}

}

return $count <= (5 \* $rate \* $rate) ? true : false;

}

public function hash($file){

if (!file\_exists($file)){

return false;

}

$height = 8 \* $this->rate;

$width = 8 \* $this->rate;

$img = imagecreatetruecolor($width, $height);

list($w, $h) = getimagesize($file);

$source = $this->createImg($file);

imagecopyresampled($img, $source, 0, 0, 0, 0, $width, $height, $w, $h);

$value = $this->getHashValue($img);

imagedestroy($img);

return $value;

}

public function getHashValue($img){

$width = imagesx($img);

$height = imagesy($img);

$total = 0;

$array = array();

for ($y=0;$y<$height;$y++){

for ($x=0;$x<$width;$x++){

$gray = ( imagecolorat($img, $x, $y) >> 8 ) & 0xFF;

if (!is\_array($array[$y])){

$array[$y] = array();

}

$array[$y][$x] = $gray;

$total += $gray;

}

}

$average = intval($total / (64 \* $this->rate \* $this->rate));

$result = '';

for ($y=0;$y<$height;$y++){

for ($x=0;$x<$width;$x++){

if ($array[$y][$x] >= $average){

$result .= '1';

}else{

$result .= '0';

}

}

}

return $result;

}

public function createImg($file){

$ext = $this->getFileExt($file);

if ($ext === 'jpeg') $ext = 'jpg';

$img = null;

switch ($ext){

case 'png' : $img = imagecreatefrompng($file);break;

case 'jpg' : $img = imagecreatefromjpeg($file);break;

case 'gif' : $img = imagecreatefromgif($file);

}

return $img;

}

public function getFileExt($file){

$infos = explode('.', $file);

$ext = strtolower($infos[count($infos) - 1]);

return $ext;

}

}