Anime Cel Pigment References   
Preservation Project (v1.1)

Technical specifications for STAC / Taiyo-Shikisai /  
 USA-Cartoon Colour Cel-Vinyl colour charts  
 including  sRGB / CMYK / PANTONE matching system.

Version 1.1 • August, 2025  
 European ICC Profile (ECI/FOGRA)  
Alexandr Gomez

***Intentionally left blank***

**Technical Colour Workflow for almost any classic Anime / Cel Pigment Charts**

**SUMMARY**

This document consolidates sources, specifications, decisions,   
and reference code to generate sRGB, CMYK samples   
and identify closest Pantone tones from CIE L\*a\*b\* values for:

\* **スタック**（*STAC：Saito Tele-Anima Colors Co. Ltd.）*  
\* **太陽色彩**（*TAIYO-SHIKISAI/ 太陽色彩株式会社 ANIMATION. PAINT）*  
\* **Cartoon Cel-Vinyl** Colour Charts

*1.* Sources  *Page 4*

*2.* ICC Profiles for printing  *Page 7*

*3.* Colour Processing  *Page 8*

*4.* Pantone Matching  *Page 11*

*5.* Quality Preservation  *Page 13*

*6.* Considerations & Final Notes  *Page 16*

Version: 1.1 • Date: 24/08/2025 • Scope: Europe (ICC ECI/FOGRA commercial printing profiles)

# 1. Sources (LAB, RGB, Hex sRGB, Hex ProPhoto, HSL)

The colour specifications originate from captures and measurements performed on pigment charts for cels (vinyl). Original dataset (with no Pantone nor CMYK conversions) was sourced from the Kanzenshuu community forum:

[**https://www.kanzenshuu.com/forum/viewtopic.php?t=19448**](http://www.kanzenshuu.com/forum/viewtopic.php?t=19448)

***A large part of this work has been based on the complete spectrophotometry carried out for that document. Unfortunately, it has not yet been possible to locate   
and contact the person who performed those measurements.***

***This project, and this document, will be updated once contact is established.***

*──────────────────────────────────────────────────────*

***- Original Measuring Details -***

|  |  |  |
| --- | --- | --- |
| **Measuring Device** | **Date Created** | **Notes** |
| Epson Perfection v600 | 2021-12-27 | 48-bit scanning, ProPhoto RGB embedded ICC profile |
| ColourMunki Photo | 2024-05-23 | Spectrophotometric remeasurement D50/2° standard observer |
| X-Rite i1Pro 2 | 2025-04-09 | Final wide-gamut capture addressing metamerism issues |

## *- Technical scanning and conversion specifications notes -*

* + Many pigment Colours exceed sRGB gamut boundaries. Accurate display requires Rec.2020 compatible monitors (limited availability as of 2021).
  + Original charts scanned at 48-bit depth, saved as TIFF with embedded ProPhoto RGB ICC profile for maximum Colour fidelity.
  + Spectrophotometric remeasurement provided device-independent CIE L\*a\*b\* values as ground truth references.
  + Derived sRGB values computed from L\*a\*b\* coordinates with gamut clipping applied when Colours fall outside sRGB boundaries.
  + Neutral reference points standardized:Closest Pantone calculation from L\*a\*b\* (DE00)  
      **BLACK at LAB(6,0,0)  sRGB(19,19,19), WHITE at LAB(95,0,0)  sRGB(240,240,240)**
  + STAC <-> Taiyo-Shikisai **conversions implemented via Toei Animation in-house lookup table** (not physically equivalent Colours).
  + Physical chart aging and yellowing affected low-saturation/high-lightness samples. Retrobright treatment applied to restore accuracy.

***- Database evolution timeline - :***

|  |  |
| --- | --- |
| **Date** | **Update Description** |
| **2025-08-24** | **Consolidation for STAC / TAIYO / USA CARTOON colour charts by including CMYK and closest Pantone values (using a PSOcoated V3 ICC colour profile).**  **All the colour charts catalogued in one document (this one!) as faithfully as possible to the original values, both in print (CMYK) and digitally on screen (sRGB).** |
| ***From here backwards, it corresponds to the original Excel work  provided in the forum*** | |
| 2022-05-18 | Integration of older STAC chart scan Colour code nomenclature updates Deprecated codes mapping established |
| 2022-07-25 | ProPhoto RGB hexadecimal values added for workflow convenience |
| 2023-01-19 | Complete X-Colour chart digitization |
| 2023-01-20 | STAC A-Colour (those created by TOEI for USA animes) chart integration |
| 2023-01-27 | sRGB HSL values computed and added for UI/web design applications |
| 2023-02-21 | STAC  Taiyo-Shikisai conversion table implemented from Toei reference |
| 2023-04-30 | Taiyo-Shikisai 595-Colour edition 8-booklet comprehensive scan |
| 2024-01-15 | Extended STAC  Taiyo-Shikisai mapping with additional missing entries |
| 2024-05-13 | STAC spectrophotometric recapture replacing scanner-based measurements |

|  |  |
| --- | --- |
| 2024-05-15 | Sun Colour spectrophotometric update scanner version retained as reference |
| 2024-05-29 | X-Colour and A-Colour post-retrobright recapture with enhanced accuracy |
| 2025-02-01 | Cel-Vinyl chart addition wide-gamut colours beyond instrument limits |
| 2025-04-09 | Comprehensive i1Pro2 recapture metamerism correction for violet/pink hues |

*──────────────────────────────────────────────────────*

# ICC profiles and European selection (ECI/FOGRA)

For European commercial offset printing on coated substrates, the workflow employs **PSO Coated V3** (European Colour Initiative) as the primary CMYK destination profile.

This profile represents current industry standards for high-quality commercial printing and supersedes historical FOGRA39 in contemporary workflows.

Screen display maintains sRGB IEC61966-2-1 ICC profile compliance for maximum device compatibility and consistent Colour reproduction across standard monitors.

## *- Technical Profile Specification -*

* + **CMYK Output Profile:** PSO Coated v3 (ECI) - European commercial coated paper standard
  + **Colour Reference Space:** CIE L\*a\*b\* D50/2° standard observer (device-independent ground truth)
  + **RGB Working Space:** sRGB IEC61966-2-1 for display, ProPhoto RGB for archival/editing
  + **Gamut Management:** Out-of-gamut Colours handled via ICC profile rendering intent (perceptual/relative Colourimetric)
  + **Alternative Substrates:** FOGRA52 (uncoated), PSO Uncoated v3 (uncoated), FOGRA51 (newsprint)

Through the original script, using the Pantone-to-Lab CSV, it is possible to choose the ICC profile in use.

**NOTE**: This will positively affect the final colour displayed in CMYK; as an advantage, it allows adapting the document to any printing house in the world that may require any other type of ICC.

# Colour Processing: Technical Decisions

The colour space conversion implements a two-stage transformation pipeline to ensure compatibility with standard imaging libraries:

(i) CIE L\*a\*b\*  sRGB intermediate conversion, followed by (ii) sRGB CMYK conversion by using **PSO Coated V3 ICC** profile.

## *- Data Preservation Strategy -*

# Original data always preserved

original\_data = {

'L': 25.4, # 1 decimal precision

'a': -1.5, # 1 decimal precision

'b': -5.0, # 1 decimal precision

'R': 55, 'G': 61, 'B': 68 # Integer values

}

# Computed data maintains separate namespace

computed\_data = {

'C': 9.41, # 2 decimal precision

'M': 5.88, # 2 decimal precision

'Y': 14.12, # 2 decimal precision

'K': 0.00 # 2 decimal precision

}

## *- Colorimetric Conversion Chain -*

def lab\_to\_cmyk(self, L, a, b):

"""

Convert LAB to CMYK using ICC profile transformation

Critical decisions:

1. Relative Colorimetric rendering intent

2. Black Point Compensation enabled

3. PSOcoated\_v3.icc for print accuracy

"""

# Validate and clamp input values

L\_val = max(0, min(100, float(L)))

a\_val = max(-128, min(127, float(a)))

b\_val = max(-128, min(127, float(b)))

# PIL LAB format conversion

lab\_image = Image.new("LAB", (1, 1))

lab\_image.putpixel((0, 0), (

int(L\_val \* 2.55), # L\* 0-100 → 0-255

int(a\_val + 128), # a\* -128..127 → 0-255

int(b\_val + 128) # b\* -128..127 → 0-255

))

# Transform using ICC profile

cmyk\_image = ImageCms.applyTransform(lab\_image, self.transform)

c, m, y, k = cmyk\_image.getpixel((0, 0))

return (c/255.0)\*100, (m/255.0)\*100, (y/255.0)\*100, (k/255.0)\*100

## *- Presicion Requeriments - Decimal precision standards -*

| Data Type | Precision | Rationale |
| --- | --- | --- |
| **LAB Values** | 1 decimal | Industry standard, adequate for visual discrimination |
| **XYZ Tri-stimulus** | 1 decimal | Matches spectrophotometer output precision |
| **RGB Values** | Integer | 8-bit color depth standard |
| **CMYK Percentages** | 2 decimals | Print industry requirement for dot gain calculations |
| **Delta E CIE2000** | 3 decimals | Critical for <1.0 imperceptible threshold detection |
| **Hex Values** | Standard | #RRGGBB format, uppercase |

## *- Presicion Requeriments - Delta E Thresholds -*

DELTA\_E\_THRESHOLDS = {

'imperceptible': 1.0, # JND (Just Noticeable Difference)

'barely\_perceptible': 3.0, # Trained observer threshold

'perceptible': 6.0, # Average observer threshold

'clearly\_visible': 10.0 # Obvious color difference

}

## *- Delta E CIE-2000 Implementation -*

The colour space conversion also implements a two-stage transformation pipeline to ensure compatibility with standard imaging libraries:

def calculate\_delta\_e\_cie2000(self, lab1, lab2):

"""

CIE2000 Delta E calculation with complete implementation

Key improvements over CIE76/CIE94:

- Accounts for blue region distortions

- Improved neutral color handling

- Perceptually uniform chroma scaling

"""

L1, a1, b1 = lab1

L2, a2, b2 = lab2

# Weighting factors (industry standard)

kL = kC = kH = 1.0

# Chroma calculation with G factor correction

C1 = math.sqrt(a1\*\*2 + b1\*\*2)

C2 = math.sqrt(a2\*\*2 + b2\*\*2)

C\_avg = (C1 + C2) / 2.0

# G factor for improved chroma scaling

G = 0.5 \* (1 - math.sqrt(C\_avg\*\*7 / (C\_avg\*\*7 + 25\*\*7)))

# Prime values with G correction

a1\_prime = a1 \* (1 + G)

a2\_prime = a2 \* (1 + G)

# Continue with CIE2000 algorithm...

# [Full implementation in color\_processor.py]

*──────────────────────────────────────────────────────*

## *- Quality Assesment Framework -*

def categorize\_delta\_e(self, delta\_e):

"""Categorize conversion quality based on Delta E"""

if delta\_e < 1.0:

return 'excellent' # Imperceptible difference

elif delta\_e < 3.0:

return 'good' # Barely perceptible

elif delta\_e < 6.0:

return 'acceptable' # Noticeable but usable

elif delta\_e < 10.0:

return 'problematic' # Clearly visible

else:

return 'unacceptable' # Significant color shift

# Closest Pantone calculation from L\*a\*b\* (DE00)

Pantone Colour matching employs CIE DE00 (CIEDE2000) Colour difference formula to quantify perceptual Colour differences between target CIE *L\*a\*b\** coordinates and Pantone Solid Coated (C) reference library.

The algorithm identifies the Pantone Colour with minimumDE00 value, providing the closest perceptual match.

CIEDE2000 (DE00) represents the most advanced Colour difference formula, incorporating corrections for lightness, chroma, and hue perception non-linearities, particularly in blue and gray regions where human visual system exhibits reduced discrimination sensitivity.

## *- Euclidean Distance in P erceptual Space -*

def find\_closest\_pantone(self, L, a, b, max\_delta\_e=None):

"""

Pantone matching using CIE2000 in LAB space

Decision rationale:

- LAB space provides perceptual uniformity

- CIE2000 handles edge cases better than CIE76

- Database of 2000+ Pantone colors with LAB values

"""

input\_lab = (float(L), float(a), float(b))

best\_match = None

min\_delta = float('inf')

for pantone in self.pantone\_database:

pantone\_lab = (pantone['L'], pantone['a'], pantone['b'])

delta\_e = self.calculate\_delta\_e\_cie2000(input\_lab, pantone\_lab)

if delta\_e < min\_delta:

min\_delta = delta\_e

best\_match = pantone

return best\_match['name'], best\_match['code'], min\_delta

*──────────────────────────────────────────────────────*

## ***- E00*** *interpretation -*

* D**E00 < 1.0:** Imperceptible difference under standard viewing conditions
* D**E00 1.0-2.0:** Barely perceptible to trained observers under optimal conditions
* D**E00 2.0-5.0:** Noticeable difference but acceptable for commercial applications
* D**E00 5.0-10.0:** Clear colour difference, requires attention in critical applications
* D**E00 > 10.0:** Distinctly different colours, unsuitable for colour matching.

*──────────────────────────────────────────────────────*

## *- Technical limitations / Accuracy considerations -*

* Some colours may exceed Pantone gamut boundaries, resulting in E00 > 5.0
* Colour matches under D50º illuminant may vary under different light sources
* Pantone physical standards have ±1.5 DE00 manufacturing tolerance.
* Fluorescent or metallic pigments cannot be accurately matched.
* Individual Colour perception differences may affect the results.

*──────────────────────────────────────────────────────*

# Conversion Quality Metrics

For professional Colour documentation r**equiring both screen display and print reproduction**, the recommended format presents dual Colour visor (sRGB and real CMYK simulation) accompanied by comprehensive technical data tables, by ensuring accurate colour communication across digital and print media workflows.

Future EPUB implementation should embed RGB Colour swatches with sRGB ICC profile metadata while maintaining numeric Colour data in accessible text format.

PDF/X standards are reserved for pre-press applications requiring embedded CMYK Colour spaces.

class QualityMetrics:

def \_\_init\_\_(self, conversion\_results):

self.cmyk\_deltas = [r['cmyk\_delta\_e00'] for r in conversion\_results]

self.pantone\_deltas = [r['pantone\_delta\_e00'] for r in conversion\_results

if r['pantone\_delta\_e00'] is not None]

def calculate\_statistics(self):

return {

'cmyk\_mean\_delta': statistics.mean(self.cmyk\_deltas),

'cmyk\_median\_delta': statistics.median(self.cmyk\_deltas),

'pantone\_mean\_delta': statistics.mean(self.pantone\_deltas),

'excellent\_rate': self.count\_by\_quality('excellent') / len(self.cmyk\_deltas),

'problematic\_rate': self.count\_by\_quality('problematic') / len(self.cmyk\_deltas)

}

## *- Expected Quality Benchmarks -*

| Metric | Target | Rationale |
| --- | --- | --- |
| **CMYK Mean ΔE** | < 3.0 | Acceptable for print production |
| **Excellent Rate** | > 60% | Majority of colors imperceptible difference |
| **Problematic Rate** | < 10% | Minimize clearly visible color shifts |
| **Pantone Match Rate** | > 80% | Industry database completeness |

***- ICC Profile Integration Method -***

# Profile Selection Rationale  
  
ICC\_PROFILE\_DECISIONS = {

'primary': 'PSOcoated\_v3.icc',

'reason': 'ISO 12647-2 standard for offset printing',

'characteristics': {

'gamut': 'Optimized for coated paper printing',

'black\_generation': 'GCR (Gray Component Replacement)',

'rendering\_intent': 'Relative Colorimetric',

'black\_point\_compensation': True

}

}

## *- Rendering Intent Analysis -*

RENDERING\_INTENTS = {

'perceptual': {

'use\_case': 'Photographic images',

'gamut\_mapping': 'Compresses entire gamut proportionally',

'decision': 'Not chosen - too much color shift for spot colors'

},

'relative\_colorimetric': {

'use\_case': 'Spot colors and logos',

'gamut\_mapping': 'Clips out-of-gamut colors to nearest equivalent',

'decision': 'CHOSEN - Preserves in-gamut colors exactly',

'black\_point': 'Compensated to avoid gray shifts'

}

}

# *- Batch Processing Strategy Adopted -*

class BatchProcessor:

def \_\_init\_\_(self, chunk\_size=100):

self.chunk\_size = chunk\_size

def process\_colors\_batch(self, color\_data):

"""

Process colors in chunks to optimize memory usage

and provide progress feedback

"""

total\_colors = len(color\_data)

processed = 0

for chunk\_start in range(0, total\_colors, self.chunk\_size):

chunk\_end = min(chunk\_start + self.chunk\_size, total\_colors)

chunk = list(color\_data.items())[chunk\_start:chunk\_end]

# Process chunk with single ICC transform initialization

for color\_id, data in chunk:

self.process\_single\_color(color\_id, data)

processed += 1

if processed % 50 == 0:

progress = (processed / total\_colors) \* 100

print(f"Progress: {progress:.1f}% ({processed}/{total\_colors})")

# *- Valitations & Testings -*

def validate\_conversion\_quality(self, results):

"""

Automated validation of conversion results

Fails pipeline if quality thresholds not met

"""

quality\_checks = {

'cmyk\_extreme\_delta': len([r for r in results if r['cmyk\_delta\_e00'] > 15]),

'missing\_pantone\_rate': len([r for r in results if r['pantone\_name'] == 'N/A']) / len(results),

'invalid\_rgb\_count': len([r for r in results if not r['has\_valid\_rgb']]),

'color\_space\_violations': self.check\_color\_space\_violations(results)

}

# Quality gates

assert quality\_checks['cmyk\_extreme\_delta'] < len(results) \* 0.05, "Too many extreme CMYK deltas"

assert quality\_checks['missing\_pantone\_rate'] < 0.3, "Pantone match rate too low"

return quality\_checks

*──────────────────────────────────────────────────────*

# 6. Considerations

## ****1. Wide Gamut Support****

* + Rec2020 color space integration for future displays
  + Extended gamut Pantone colors

## ****2. Machine Learning Enhancement****

* + Trained models for better Pantone matching
  + Metameric color prediction

## ****3. Real-time Processing****

* + GPU acceleration for ICC transformations
  + Streaming pipeline for large datasets

*──────────────────────────────────────────────────────*

## *- Quality Assurance and Validation Requirements -*

* + Verify sRGB display under D65º illuminant, validate CMYK proofs under D50º.
  + Document out-of-gamut colours are noticed with indications.
  + Check the colour reproduction across different operating systems and displays.
  + Include textual Colour descriptions and num. values for vision-impaired users.
  + Colours exceeding sRGB/CMYK gamut boundaries   
    undergo automatic clipping with potential Colour shift.
  + Results are computational approximations;   
    physical verification with Pantone Colour Bridge required.
  + Colour appearance depends on illuminant and observer;   
    maintain D50/2° reference standard.
  + Commercial printing introduces ±2-3 E00 variation;   
    specify acceptable tolerance ranges.

## *****- Final Notes -*****

While the data capture has been carried out under the best possible technical conditions, this document cannot exactly substitute the precise colorimetric compositions used by the manufacturers. In fact, not even the manufacturers themselves could maintain 100% colorimetric accuracy when producing their pigments.

In any case, the aim of this project is to faithfully preserve the composition of   
each of the pigments that the animation industry presented to us.

If you want to preserve these colour charts in printed form, you can use your own ICC profiles (Remember to explain to your printer that the colours must not be converted, but rather the embedded ICC profile in the document must be used).

# 7. References

* <https://www.kanzenshuu.com/forum/viewtopic.php?t=19448>
* <http://www.style.fm/as/05_column/tsujita/tsujita_bn.shtml>
* <https://animestyle.jp/column/>
* <https://www.nekomataya.info/>
* **CIE Publication 15:2004** - Colorimetry, 3rd Edition
* **ISO 12647-2:2013** - Process control for offset lithographic processes
* **CIE Technical Report 224:2017** - Colour fidelity index for accurate scientific use
* **ICC.1:2010** - Image technology colour management — Architecture, profile format and data structure

— End of Technical Documentation —

*──────────────────────────────────────────────────────*

***Work done out of love for colour   
and preservation of hand-made   
anime material and techniques !***