
**Image technology colour
management — Black point
compensation**

*Gestion de couleur en technologie d'image — Compensation du point
noir*

STANDARDSISO.COM : Click to view the full PDF of ISO 18619:2015



STANDARDSISO.COM : Click to view the full PDF of ISO 18619:2015



COPYRIGHT PROTECTED DOCUMENT

© ISO 2015, Published in Switzerland

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword	iv
Introduction	v
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Requirements	2
4.1 Constraints.....	2
4.2 Computation.....	3
4.2.1 Outline.....	3
4.2.2 Functions used.....	3
4.2.3 Computing the SourceBlackPoint.....	4
4.2.4 Computing the DestinationBlackPoint for ICC profiles that are not LUT-based.....	4
4.2.5 Computing the DestinationBlackPoint for ICC profiles that are LUT-based.....	5
4.2.6 Computing the mapping from SourceBlackPoint to DestinationBlackPoint.....	10
4.2.7 Applying the black point compensation in a colour conversion.....	11
Annex A (informative) Why black point compensation is necessary	12
Bibliography	14

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the WTO principles in the Technical Barriers to Trade (TBT) see the following URL: [Foreword - Supplementary information](#)

The committee responsible for this document is ISO/TC 130, *Graphic technology*, in cooperation with the International Color Consortium.

Introduction

Black point compensation (BPC) is a technique used to address colour conversion problems caused by differences between the darkest level of black achievable on one device and the darkest level of black achievable on another. This procedure was first implemented in Adobe Photoshop in the late 1990s. The International Color Consortium (ICC) and ISO Technical Committee 130 (Graphic technology) have created this document to allow black point compensation to be used in a consistent manner across applications.

The purpose of BPC is to adjust a colour transform between the colour spaces of source and destination ICC profiles, so that it retains shadow details and utilizes available black levels. The procedure depends only on the rendering intent(s) and the source and destination ICC profiles, not on any points in a particular image. Therefore, the colour transform using specific source and destination ICC profiles and rendering intent can be computed once, and then efficiently applied to many images which use the same ICC profile colour transform pair and rendering intent.

STANDARDSISO.COM : Click to view the full PDF of ISO 18619:2015

STANDARDSISO.COM : Click to view the full PDF of ISO 18619:2015

Image technology colour management — Black point compensation

1 Scope

This International Standard specifies a procedure, including computation, by which a transform between ICC profiles can be adjusted (compensated) to take into account differences between the dark end of the source colour space and the dark end of the destination colour space. This is referred to as black point compensation (BPC). The relative colorimetric encoding of ICC profile transforms already provides a mechanism for such adjustment of the light (white) end of the tone scale.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 15076-1, *Image technology colour management — Architecture, profile format and data structure — Part 1: Based on ICC.1:2010*

ICC.1:2001-04, *File Format for Color Profiles*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 15076-1 and the following apply.

3.1

DestinationBlackPoint

coordinate representing a dark neutral reproducible colour in the destination colour gamut

3.2

DestinationProfile

ICC profile, containing the transform from profile connection space to the destination device colour space

3.3

SourceBlackPoint

coordinate representing a dark neutral colour in the source colour gamut

3.4

SourceProfile

ICC profile, containing the transform from the source device colour space to the profile connection space

3.5

RenderingIntent

rendering intent of the conversion from a source ICC profile's colour space to a destination ICC profile's colour space

3.6

LabIdentityProfile

real or virtual ICC profile that contains a bi-directional (identity) transform between CIELAB and PCSLAB

3.7

black point compensation

BPC

computational procedure by which a transform between the colour spaces of ICC profiles can be adjusted (compensated) to take into account differences between the dark end of the source colour space and the dark end of the destination colour space

3.8

L, a, b

L^* , a^* , or b^* component of the CIELAB colour space

3.9

output-capable CMYK profile

CMYK profile containing a transform from the ICC PCS encoding to the colour space encoding

3.10

transform

mathematical operations that define the change in representation of a colour between two colour spaces

3.11

gamut

range of colours that a given system is capable of reproducing

4 Requirements

4.1 Constraints

The black point compensation procedure defined in this International Specification shall take as its inputs a destination ICC profile, a source ICC profile, and a rendering intent (in this International Standard called *DestinationProfile*, *SourceProfile* and *RenderingIntent* respectively).

Applications that apply black point compensation shall support ICC profiles that conform to ISO 15076-1 and ICC profiles that conform to ICC.1:2001-04.

NOTE 1 This requirement ensures that processing applications will properly process all Version 2 and Version 4 ICC profiles.

NOTE 2 ISO 15076-1 provides a description of source and destination ICC profiles.

The rendering intent shall be one of: RelativeColorimetric; Perceptual; or Saturation. The rendering intent used with *DestinationProfile* shall be the same as the rendering intent used with *SourceProfile*. Black point compensation is not appropriate for the AbsoluteColorimetric rendering intent.

The versions of *SourceProfile* and *DestinationProfile* do not need to match.

SourceProfile and *DestinationProfile* types shall be Input, Display, Output, or ColorSpace. The types of *SourceProfile* and *DestinationProfile* do not need to match.

DestinationProfile shall contain a transform from the ICC PCS encoding to the colour space encoding.

The data colour spaces of the *SourceProfile* and *DestinationProfile* shall be Gray, RGB, CMYK or CIELAB. The data colour spaces of *SourceProfile* and *DestinationProfile* do not need to match.

NOTE 3 Some implementations of BPC extend to additional colour spaces or mixed rendering intents. The way in which BPC operates in these cases is outside the scope of this International Standard.

4.2 Computation

4.2.1 Outline

Black point compensation shall be performed according to the following procedure:

- a) the *SourceBlackPoint* of *SourceProfile* shall be calculated as specified in 4.2.3;
- b) the *DestinationBlackPoint* of *DestinationProfile* shall be calculated as specified in 4.2.4 and 4.2.5;
- c) a mapping from *SourceBlackPoint* to *DestinationBlackPoint* shall be calculated as specified in 4.2.6;
- d) the mapping shall be applied in a colour conversion as specified in 4.2.7.

4.2.2 Functions used

4.2.2.1 Colour transform with profiles

T shall denote a function to transform a point in the data colour space of *Profile1* to a point in the device colour space of *Profile2*, using a rendering intent, such that

$$y = \mathbf{T}(x, \text{Profile1}, \text{Profile2}, \text{Intent}) \quad (1)$$

where

x is a point in the data colour space of *Profile1*;

y is a point in the data colour space of *Profile2*;

Intent is the rendering intent.

4.2.2.2 Darkest colour of a profile

D shall denote a function to provide the darkest colour coordinate in the data colour space of *Profile* for a rendering intent, such that

$$dc = \mathbf{D}(\text{Profile}, \text{Intent}) \quad (2)$$

where

dc is the darkest colour

Profile is the profile being evaluated,

Intent is the rendering intent.

dc shall be determined as follows:

A subset of the vertices within the data colour space of *Profile*, *V*, shall be defined as follows.

If the data colour space of *Profile* is Gray

V shall be the set of {(0) (1)}.

If the data colour space of *Profile* is RGB

V shall be the set of {(0, 0, 0) (1, 1, 1)}.

If the data colour space of *Profile* is CMYK

V shall be the set of {(0, 0, 0, 0) (1, 1, 1, 1) (0, 0, 0, 1) (1, 1, 1, 0)}.

The darkest colour, *dc*, shall be the lowest value of L^* resulting from applying each element *v* of the appropriate *V* through the following transform:

$T(v, Profile, LabIdentityProfile, Intent)$.

NOTE 1 *dc* is intended to be the value of black or darkest colour in the colour space of *Profile*.

NOTE 2 Determining the darkest colour in this way works for profiles with both the normal polarity and inverse polarity.

4.2.3 Computing the SourceBlackPoint

The *SourceBlackPoint* is computed by first defining *LocalBlack* of *SourceProfile* and then using this to compute the *SourceBlackPoint*.

A *LocalBlack* value for the source colour space shall be defined as follows:

If *SourceProfile* is an output-capable CMYK ICC profile

LocalBlack shall be set to $T((0, 0, 0), LabIdentityProfile, SourceProfile, Perceptual)$

If *SourceProfile* is not an output-capable CMYK ICC profile

If the data colour space of *SourceProfile* is CIELAB,

LocalBlack shall be set to (0, 0, 0).

If the data colour space of *SourceProfile* is Gray, RGB, or CMYK,

LocalBlack shall be set to $D(SourceProfile, RenderingIntent)$.

The *SourceBlackPoint* is then calculated as follows

Li shall be set to the L^* component of $T(LocalBlack, SourceProfile, LabIdentityProfile, RenderingIntent)$.

If *Li* is greater than 50,

SourceBlackPoint shall be set to (50, 0, 0),

else

SourceBlackPoint shall be set to (*Li*, 0, 0).

4.2.4 Computing the DestinationBlackPoint for ICC profiles that are not LUT-based

If *DestinationProfile* is not LUT-based, then its *DestinationBlackPoint* shall be determined as follows:

A *LocalBlack* value for the *DestinationProfile* data destination colour space shall be determined as follows:

LocalBlack shall be set to $D(DestinationProfile, RenderingIntent)$.

NOTE Only Gray and RGB are valid data colour spaces for a *DestinationProfile* that is not LUT-based.

The *DestinationBlackPoint* is then calculated as follows

Li shall be set to the *L* component of $T(LocalBlack, DestinationProfile, LabIdentityProfile, RenderingIntent)$.

If *Li* is greater than 50,

DestinationBlackPoint shall be set to (50, 0, 0),

else

DestinationBlackPoint shall be set to (*Li*, 0, 0).

4.2.5 Computing the *DestinationBlackPoint* for ICC profiles that are LUT-based

4.2.5.1 Overview

If *DestinationProfile* is LUT-based, then its *DestinationBlackPoint* shall be determined as follows:

InitialLab, *inRamp*, and *outRamp* shall be calculated as specified in [4.2.5.2](#).

If *outRamp* is not valid as specified in [4.2.5.3](#),

DestinationBlackPoint shall be set to (0, 0, 0),

else if *RenderingIntent* is RelativeColorimetric, and the *outRamp* meets the mid range straight test specified in [4.2.5.4](#),

DestinationBlackPoint shall be set to *InitialLab*,

else

DestinationBlackPoint shall be determined as specified in [4.2.5.5](#).

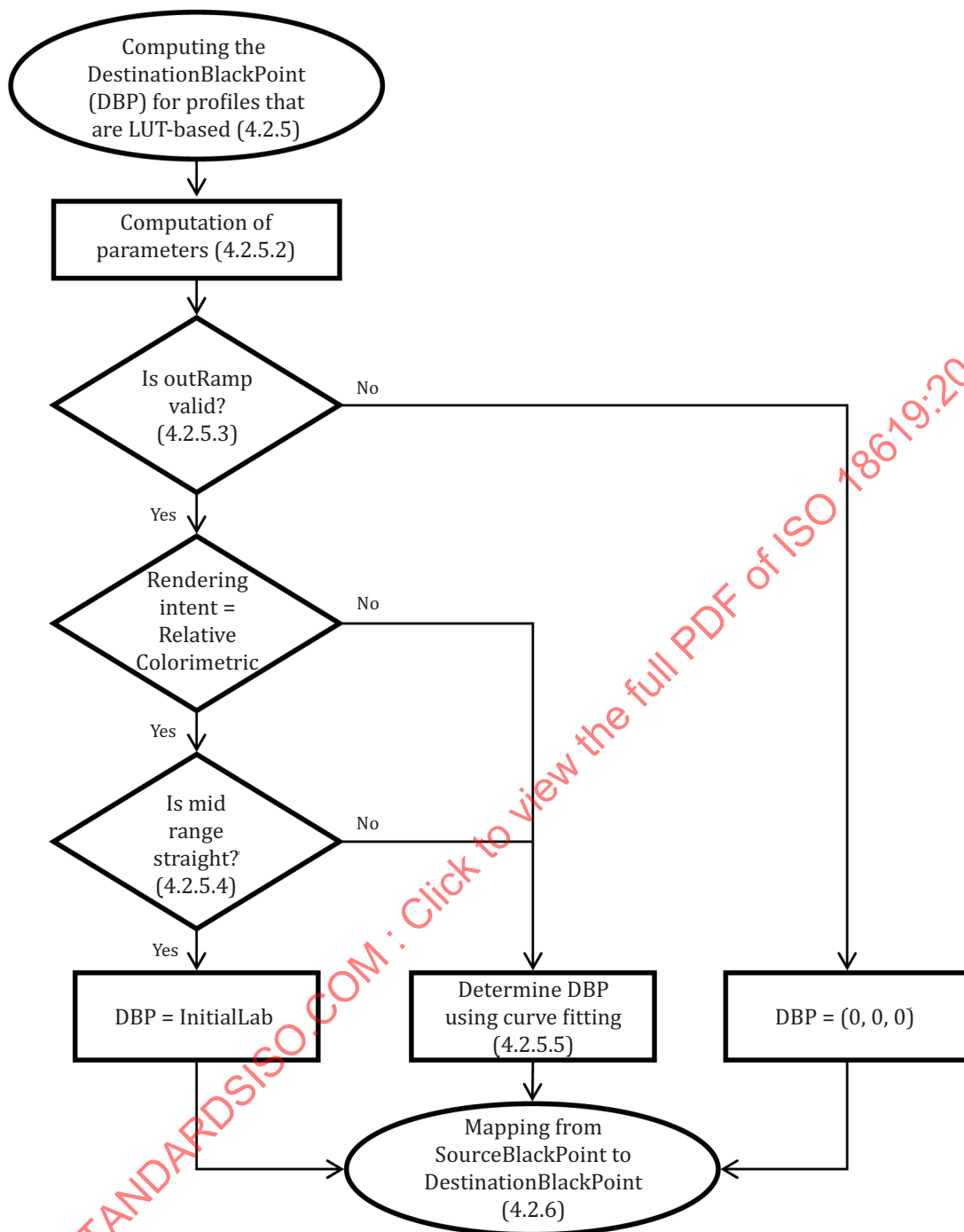


Figure 1 — DestinationBlackPoint computation logic sequence

4.2.5.2 Computation of parameters

4.2.5.2.1 InitialLAB Calculation

If *RenderingIntent* is not *RelativeColorimetric*, then *InitialLab* shall be set to (0,0,0), else *InitialLab* shall be calculated as follows:

If the colour space of *DestinationProfile* is CMYK

LocalBlack shall be set to **T** ((0,0,0), *LabIdentityProfile*, *DestinationProfile*, *Perceptual*)

If the data colour space of *DestinationProfile* is CIELAB,

LocalBlack shall be set to (0,0,0).

If the data colour space of *DestinationProfile* is Gray, or RGB,

LocalBlack shall be set to **D** (*DestinationProfile*, *RenderingIntent*).

InitialLab shall be set to **T** (*LocalBlack*, *DestinationProfile*, *LabIdentityProfile*, *RenderingIntent*).

If the data colour space of *DestinationProfile* is CMYK, then the *a** and *b** components of *InitialLab* shall be set to 0.

If the *L** of *InitialLab* exceeds 50, then the *L** component of *InitialLab* shall be set to 50.

4.2.5.2.2 Calculation of inRamp

inRamp (a list of *L** values) shall be calculated as follows:

ka shall be min(50, max(-50, *a** component of *InitialLab*),

kb shall be min(50, max(-50, *b** component of *InitialLab*),

inRampLab shall be a list of Lab values interpolated from (0, *ka*, *kb*) to (100, 0, 0) in 256 equal steps,

inRamp shall be a list of the *L** component values from *inRampLab*.

4.2.5.2.3 Calculation of outRamp

BT is a transform that converts CIELAB colour values from CIELAB space to destination device colour space to CIELAB space, as shown in [Figure 2](#).

BT (*x*) shall be defined as

$$\mathbf{T}(y, \text{DestinationProfile}, \text{LabIdentityProfile}, \text{RelativeColorimetric}) \quad (3)$$

where

x is a CIELAB colour value;

y is the result of the transform **T** (*x*, *LabIdentityProfile*, *DestinationProfile*, *RenderingIntent*);

LabIdentityProfile is as defined in [3.6](#);

DestinationProfile and *RenderingIntent* are as defined in [4.1](#).

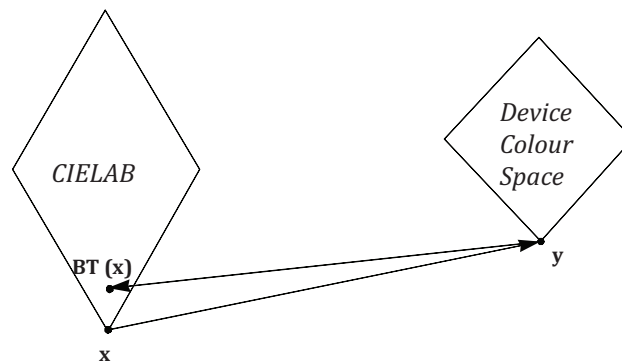


Figure 2 — BT transform

outRampLab shall be set to a list of transformed Lab values found by applying BT to each of the elements of *inRampLab*

outRamp shall be set to a list of the L^* components of *outRampLab*.

If *outRamp* is not monotonically increasing, its values shall be modified to make the list monotonically increasing. This adjustment shall not change the *last* (lightest) value in the list. This can be accomplished by looping from the second to last element to the *first* element of *outRamp* and setting the i^{th} element to the minimum of the i^{th} element and the element just after the i^{th} element. This is shown in the following pseudo code:

for ($i = (\text{last} - 1)$ to *first*)

$\text{outRamp}[i] = \min (\text{outRamp}[i], \text{outRamp}[i+1])$ (4)

where

first indicates the first index of *outRamp*;

last identifies the last index of *outRamp*.

4.2.5.3 Test for *outRamp* valid

When *outRamp*[*first*] is not less than *outRamp*[*last*] then the *outRamp* is considered invalid and the *DestinationBlackPoint* shall be set to (0, 0, 0).

4.2.5.4 Test for *midrange* straight

If *RenderingIntent* is RelativeColorimetric and the *outRamp* is valid as determined by 4.2.5.3, then the following test shall be applied, otherwise *DestinationBlackPoint* shall be determined using curve fitting as defined by 4.2.5.5.

The test for *midrange* being straight is as follows:

Assume that *midrange* is straight

MinL shall be the value of the first element of *outRamp*

MaxL shall be the value of the last element of *outRamp*

For all index values i in *inRamp* and *outRamp*

If $((\text{inRamp}[i] > \text{MinL} + 0,2 \times (\text{MaxL} - \text{MinL}))$ and

(**abs** ($inRamp[i] - outRamp[i]$) ≥ 4))

then

midrange is not straight.

If *midrange* is straight (as determined above) then the *DestinationBlackPoint* shall be the same as *InitialLab*. Otherwise, the *DestinationBlackPoint* shall be determined as specified in 4.2.5.5.

4.2.5.5 Computation of DestinationBlackPoint using curve fitting

This sub-clause defines how to calculate *DestinationBlackPoint* using curve fitting of *inRamp* and *outRamp*.

yRamp shall be set to a list of values:

$$yRamp[i] = (outRamp[i] - MinL) / (MaxL - MinL) \quad (5)$$

for all valid index values *i*

where

MinL = *outRamp*(first);

MaxL = *outRamp*(last).

The shadow curve points, $SP = \{ (inRamp[i], yRamp[i]) \}$, shall define a set of $(inRamp[i], yRamp[i])$ points for limited range of index values of *i* as follows:

If *RenderingIntent* is RelativeColorimetric then shadow points are points defined by indices *i* where $(0,1 \leq yRamp[i] < 0,5)$,

Otherwise shadow points are points defined by indices *i* where $(0,03 \leq yRamp[i] < 0,25)$.

If there are fewer than 3 points in *SP* then curve fitting cannot be applied to the data, and the *DestinationBlackPoint* shall be set to (0, 0, 0).

Otherwise the scalar values *a*, *b*, *c*, shall be determined by using a least-squares error algorithm to fit a quadratic curve of the following form:

$$y = ax^2 + bx + c \quad (6)$$

using the shadow curve points, *SP*, as the set of input values to the algorithm.

A value of *z* shall be determined to define the *DestinationBlackPoint* as follows:

If (**abs**(*a*) < 1 E-10)

z shall be set to **max** (0, **min** (50, $-c / b$)),

else

d shall be set to $(b^2 - 4 a c)$.

if ($d < = 0$)

z shall be set to 0,

else

z shall be set to **max** (0, **min** (50, $(-b + \text{sqrt}(d)) / (2 a)$)).

NOTE 1 z has been assigned to the root of the quadratic with the positive gradient. This is either the upper or lower root depending on the sign of a .

The *DestinationBlackPoint* shall be set to $(z, 0, 0)$.

Figure 3 shows an example of estimating the *DestinationBlackPoint* using curve fitting. This figure shows an example *outRamp*, the range of shadow points, the fitted quadratic curve, and the *DestinationBlackPoint*.

NOTE 2 Curve fitting is actually performed using *yRamp* which is a scaled version of *outRamp*.

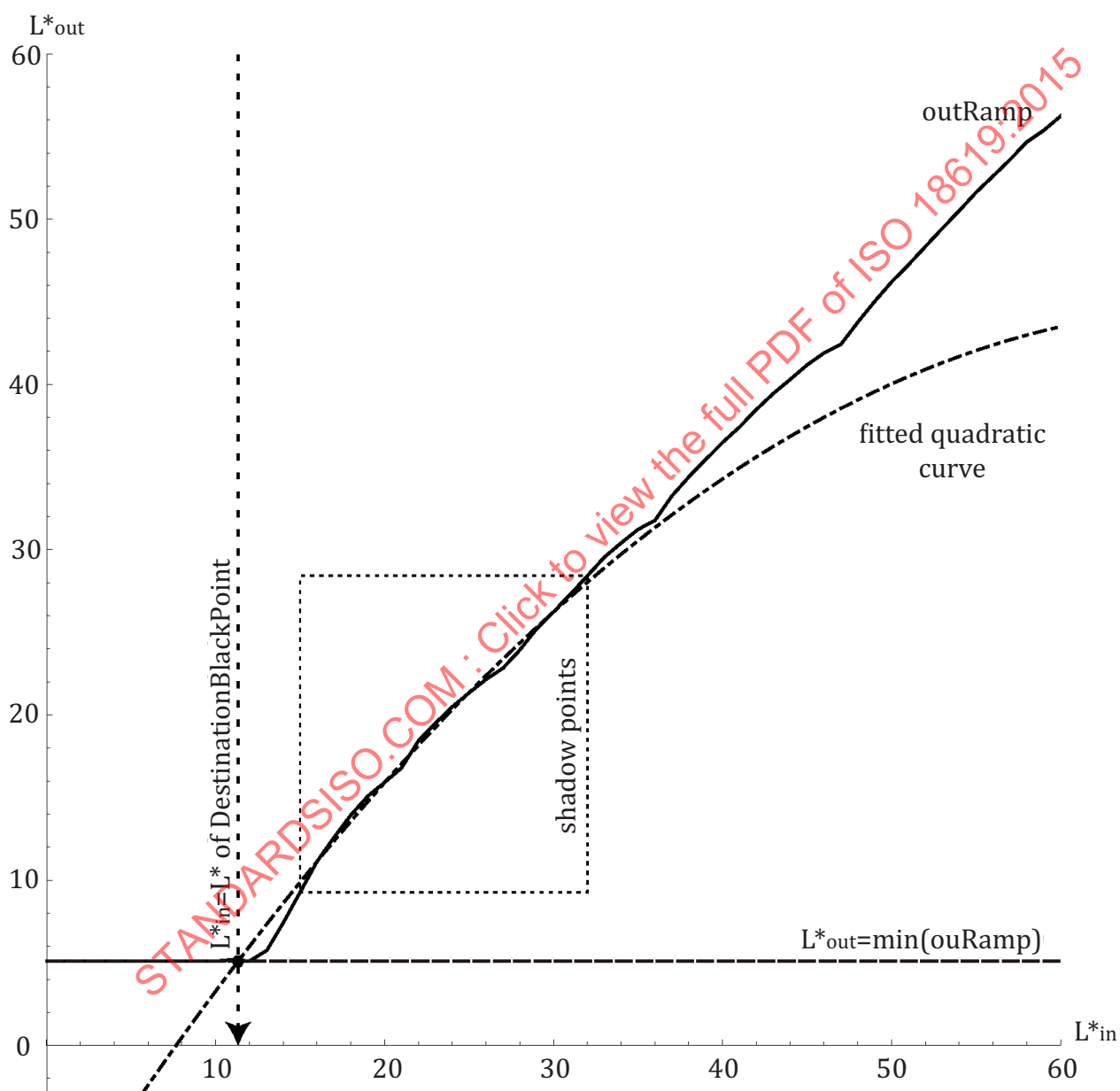


Figure 3 — Example Curve Fitting

4.2.6 Computing the mapping from *SourceBlackPoint* to *DestinationBlackPoint*

The mapping from *SourceBlackPoint* to *DestinationBlackPoint* shall be accomplished as follows:

Only the L^* components of the *SourceBlackPoint* and *DestinationBlackPoint* values are used, while the a^* and b^* components are ignored.

Y_{DST} shall be obtained by converting the L^* component of *DestinationBlackPoint* to a PCSXYZ Y value using the formula below,

Y_{SRC} shall be obtained by converting the L^* component of *SourceBlackPoint* to a PCSXYZ Y value using the formula below:

$scaleXYZ$ shall be set to $(1 - Y_{DST}) / (1 - Y_{SRC})$,

$offsetXYZ$ shall be set to $(1 - scaleXYZ) * W$.

where

$W = (0,9642; 1,0000; 0,8249)$, which is the PCSXYZ white point,

and

$Y = ((L + 16) / 116)^3$, For $L > 8$

$Y = L \times (((8 + 16) / 116)^3) / 8$, for $0 \leq L \leq 8$

4.2.7 Applying the black point compensation in a colour conversion

Black point compensation is applied at the point in the colour transforms where output results from applying *SourceProfile* are used as the input for applying *DestinationProfile* as follows:

The *SourceProfile* data colour space value shall be converted to PCSXYZ using *SourceProfile* according to *RenderingIntent*, (including conversion from PCSLAB to PCSXYZ when the ICC profile's PCS is PCSLAB). XYZ_{SRC} shall be the PCSXYZ value resulting from this conversion.

A black point compensated PCSXYZ value (XYZ_{DST}) shall be calculated using the Formula (7):

$$XYZ_{DST} = scaleXYZ * XYZ_{SRC} + offsetXYZ \quad (7)$$

The XYZ_{DST} value shall be converted to device colour space using the *DestinationProfile*, including conversion from PCSXYZ to PCSLAB when the ICC profile's PCS is PCSLAB.