

デジタル写真資料の WWW に基づく共有・管理手法について

How to share and maintain archives of digital photographs on the WWW

守岡 知彦

MORIOKA, Tomohiko

京都大学人文科学研究所

〒 606-8265 京都市左京区北白川東小倉町 47

Institute for Research in Humanities, Kyoto University

47 Kitashirakawa-Higashioguramachi, Sakyo ward, Kyoto, 606-8265

概要

大量のデジタル写真資料を少ない手間で共有・整理・活用するための方策について検討する。複数人で情報を共有するには WWW ベースで行うのが良いが、このためにはバッチ処理で各種 RAW や JPEG 等から変換処理を行ったりメタデータを抽出する必要がある。また、撮影データではなく、より意味的な情報の付与・編集をどのように（なるべく手間をかけず）行うかも重要である。このために現在取り得る手段や見通しについて議論したい。

Abstract

In this paper, we will discuss how we should share/arrange/use a lot of digital photographs without a lot of time and labor. To share various information among plural members, it is good way to use WWW. However it requires batch processing for various digital photo data, such as conversion from various kind of "RAW format" to JPEG, and reading of various metadata from RAW data or JPEG/Exif. In addition, we don't need only automatically recorded metadata, but also more semantic information: so it is important how we add/edit semantic annotations of photographs easily. In this paper, we will discuss available methods to treat RAW data and semantic annotations, and possible models and goals to develop archives of digital photographs.

キーワード： デジタル写真、RAW データ、Wiki、アノテーション、データベース

Keywords: digital photography, RAW data, Wiki, annotation, database

1 はじめに

デジタル写真は撮影や処理が容易で、ランニングコストも安く、低コストで大量の写真撮影することが可能である。このため、視覚的な情報を扱うような人文学・人文情報学のさまざまな分野で有用な道具となっている。しかしながら、低コストで大量の写真が撮影できてしまうということ

は、大量の写真データを扱わなければならないということを意味しており、データの管理や整理、処理等に大きな手間がかかり、せつかくのデータが死蔵され、うまく活用されにくいということにもなりかねない危険性もはらんでいるといえる。

こうした問題を解決するひとつの方法は、WWW を用いて複数人でデータを共有し、Wiki 的な手法でもって共同でデータを整理することではないかと思われる。しかしながら、デジタル写真の画像データの場合、テキストデータの場合とはまた別の問題がある。

ひとつは画素数の問題である。今の一般的なデジタル一眼レフカメラの画素数は、エントリーモデルも含めて、通常、1000 万画素以上であるが、XGA で 80 万画素弱、今の一般的なノートパソコン (WXGA) だと通常 100 万画素程度、一般的なデスクトップパソコン (full HD/WUXGA) でも 200 万画素程度であり、撮影されたデータの画素数に比べてはるかに少ない。

また、色の分解能も一般的なデジタル一眼レフカメラでは RGB それぞれについて 10~14bit 程度あるのに対し、一般的なパソコンではそれぞれ 8bit 以上を扱うことができなかつたり困難だつたりする。表現可能な色域に関しても、一般的なデジタル一眼レフカメラでは AdobeRGB (拡張色空間) が扱えるものが多いが、Windows + IE の場合、sRGB 以外の色空間を扱うことができない。標準的な画像フォーマットである JPEG では、色空間に関しては Exif や ICC プロファイルを用いることで AdobeRGB やその他各種色空間を用いることができるが、色の分解能に関しては 8bit に制限されている。このように、デジタルカメラやディスプレイ、プリンター等の入出力機器の性能向上に比べて、OS や WWW ブラウザー、標準的な画像配布用フォーマットの側の対応が遅れているのが実情であり、オリジナルデータそのままでは WWW 上で扱いつらく、利用者の環境に合わせて適切に変換してやる必要がある。それでも、色を適切に再現するのは必ずしも容易ではなく『色化け』が起こる。『色化け』は文字化けと違って気づきにくく、その点で問題である。

センサーの能力を十分に活かすためには JPEG では不十分であり、デジタル一眼レフカメラや一部のコンパクト・デジタルカメラでは『RAW 形式』と呼ばれるセンサーの出力を基本的にそのまま記録 (圧縮するとしても可逆圧縮を用い、非可逆圧縮は行わない。また、ベイヤー配列のようなセンサーの特性・形状に依存した記録を行い、ピクセルへの展開は行わないなど) する画像フォーマットで撮影した画像を保存することができるようになっている。ただ、この RAW 形式は基本的にカメラ毎に異なった形式になっており、互いに互換性がない。¹

ところで、デジタル写真には撮影時の情報が記録されており、データベースとして利用する上で有用である。しかしながら、こうしたメタデータの幾つかは Exif で標準化されているものの、非標準の形式で記録される情報もあり、Exif で標準化されているような情報であってもベンダー固有の情報として記録されたり、RAW 形式でないと記録されないという場合もある。

このように、カメラで記録された情報を全て利用しようとする RAW 形式で記録する必要があるが、そうすると相互運用性が低くなり、相互運用性を高めるために標準的な JPEG/Exif の範囲内で利用すると取り出せる情報が減ってしまうというジレンマに直面することになる訳である。WWW ベースで写真を管理しようとするなら、GNU/Linux ベースのサーバーなどでバッチ処理可能なプログラムが存在している必要があるが、ここではどの程度の情報が利用可能であるかについて簡単にまとめた。ただ、実例の詳細に関しては著者が保有している Nikon D700 と Canon EOS 5D の 2 機種に限定されてしまい、網羅性を欠くことをあらかじめお断りしておく。

一方、デジタル写真をデータベースとして活用する場合、カメラが自動的に記録したメタデータだけでは不十分であり、ある意図に基づいて整理を行ったり、再配置したり、加工した上で、なんらかのアノテーションを付与する必要があるといえる。カメラが自動的に記録したメタデータは基

¹Adobe は DNG という RAW 形式のための標準形式を提唱しているが、今の所採用しているカメラは少なく、普及しているとはいいがたいのが現状である。

本的に機材や焦点距離、露出、ホワイトバランス、撮影時刻、経度・緯度（GPS がある場合）等の撮影時（のカメラ）に関する情報であり、被写体や撮影者の意図、写真が意味するものなどについての情報は無い訳である。こうした情報の幾つかはパターン認識技術等を用いることで抽出できたとしても、抽出できた情報にも誤りは含み得るし、無論、（ある程度の精度で）機械的に抽出することが難しいものもある。であるからこそ、人手による整理や編集が必要なのであり、そのために Wiki 的な手法を用いようという訳であるが、そのためには、何を対象にどういった種類の情報をどういう風に扱うかということについてモデル化する必要があるといえ、ここではこの問題に関しても議論したい。

2 写真をどうやって管理するか

2.1 日付と時刻の利用

写真を WWW ベースで管理するという事は、オリジナルの写真やさまざまに加工された写真、写真にまつわるさまざまなメタデータ等に URI を付けることだという風にとらえることができる。URI はなるべく安定的で変化しないことが望ましいので、Wiki 的な手法で編集可能にする場合、なるべく変化しにくいものをベースに変化しやすいものを属性等として付与するのが良いといえる。こうした観点で考えれば、『撮影された写真』に対して固有の ID を付与し、その ID 自体は固定するのが良いといえる。

撮影者が1人、ないしは、カメラが1台であれば、この ID として撮影時刻を用いることが考えられる。撮影時刻は機種を問わず標準的に利用可能なメタデータのひとつであり、整理する上でも判りやすいといえる。ただ、通常、時刻の分解能は秒であり、ファイル名に記録されている番号等を併用する必要がある。

カメラが複数台あり、撮影者も複数人いる場合、カメラや撮影者に ID を振ることが考えられる。ただ、カメラの ID を標準的に利用可能な情報から得ることはあまり容易ではない。カメラのベンダーやモデルに関する情報は取れるのであるが、シリアル番号の記録の仕方はベンダー（カメラ）毎に異なっており、同じモデルのカメラが複数台あった場合にはうまく区別できないかも知れない。現実的には、撮影時刻とファイル名を併用すればそうやって作った文字列が衝突することはまれかも知れないが、原理的には衝突し得るので、あまり望ましい方法と言えない。こうしたことを鑑みれば、もし、一人の撮影者が同時に複数台のカメラで撮影することはないと仮定できるならば、撮影者の側に ID を振った方が無難であると思われる。これは写真のアップロードを撮影者が行うという風にワークフローを決めることで自然に実現できると考えられる。

人間にとっての判りやすさを考慮すれば、一人の撮影者が撮影したあるまとまった一連の撮影に対して、なんらかの判りやすいシンボルを付けておくとも良いかも知れない。それが不要であれば日付だけでも良い。シンボルを付与する場合でも、日付はあった方が良いと考えられる。

そうすると、結局、

ユーザー名 + 撮影日 + 付加シンボル + 撮影時刻 + ファイル名

という組で写真に固有の ID を付与することができる。

これは例えば、

`$prefix/<ユーザー名>/<撮影年>/<月>-<日>_<付加シンボル>/<撮影時刻><ファイル名>`

という階層的ファイル名で表現することができ、URL 化することができる。

例えば、2010年9月28日18時01分38秒に撮影された写真_DSC1249.NEFの場合、ユーザー名を“tomo”、付加シンボルを“CIEAS”とした場合、

```
/opt/pages/tomo/2010/09-28_CIEAS/180138_DSC1249.NEF
```

というファイル名を付けることができ、サーバー名を photo.server とすれば、

```
http://photo.server/~tomo/2010/09-28_CIEAS/180138_DSC1249.NEF
```

というような URL に対応付けることができる。

2.2 『フィルムロール』

デジタル写真は大量の写真を低コストで容易に撮影可能であるために、撮影日単位に整理した場合、ひとつのフォルダー（ディレクトリー）に数百枚以上の写真が並んでしまうことも少なくない。しかしながら、写真の処理は時間がかかるので、ひとつのフォルダーの写真の数は50～100枚程度に押えた方が無難であるといえる。これは人間にとっての見やすさという点でも同様である。

幾つかの写真管理ソフト・現像ソフト等では、銀塩フィルム写真のアナロジーに基づいて、『フィルムロール』というような単位を導入しているものがある。すなわち、数十枚程度の一連の写真の集合をかつてのフィルムのようなものに見立てる訳である。

この一連の写真の集合はなるべくなら意味のある単位であって欲しいが、デジタル写真の場合、実際に24枚や36枚とかからなるフィルムを使っている訳ではないから、その切り方は恣意的にならざるを得ない。これは言い換えれば、安定的な情報となりにくいということの意味している。逆に、例えば、50枚毎に機械的に分けた場合、ある一連の写真が別のフィルムロールに分かれてしまう恐れがある。人間にとっての判りやすさを考慮すれば、同じような写真やある一連の写真の繋がりはなるべく同じフィルムロールに入って欲しいといえる。とはいえ、人間にとっての意味を重視するあまり、何度もやり直したくなるような単位にすると、ID/URLとして不安定になってしまうし、また、整理のための手間も増えてしまうといえる。

こうしたことを考えた場合、ある程度機械的に決定可能で、かつ、人間にとってある程度判りやすい分け方が良いといえる。問題はそのような分け方があるかということであるが、もし、短時間に連続して撮影された写真は似たようなものを撮っているか意味のある一連のシーケンスであると仮定できるならば、撮影時刻の間隔が一定以上（例えば、10分とか5分とか1分とか）になっている箇所を『切れ目』と看做して分割するという手法が考えられる。この分割は段階的に行うことも可能である。

例えば、まず1時間で分割し、分割されたロール中の写真の数が100枚を越えていたら、各ロールを30分で分割する。それでも、ロール中の写真の数が100枚を越えていたら、次は15分で分割する、という風に、分割されたロール中の写真の枚数が一定数以下になるまで切れ目の間隔を短くするという方法が考えられる。あるいは、まず全ての間隔を調べておいて、適切な間隔を推測するという手法も考えられるだろう。ちなみに著者の場合、1分程度で分割すると良い場合が多いようである。

分割されたフィルムロールはそこに含まれる最初に撮影された写真の撮影時刻（開始時刻）と最後に撮影された写真の撮影時刻（終了時刻）の組で管理するのが安定的でかつ判りやすいといえる。フィルムロールに人間にとって判りやすい名前を付ける場合、フィルムロールの属性として扱うのが良いと考えられる。

ただ、いずれにせよ、写真を同定するという観点でいえば、フィルムロールは冗長な単位であり、あるロール中の写真の数を押えるための中間的なものであるといえる。

2.3 『アルバム』

多数の写真を撮影した場合、通常、撮影した写真の全てを提示・展示するのではなく、ある意図に基づいて少数の写真を選び、それを必要に応じて加工して出力することになる。HDR (High Dynamic Range) 写真やパノラマ写真のように、複数の写真を合成して1枚の写真を作ることも考えられるが、いずれにせよ、オリジナルの写真の全てをそのまま出すのではなく、セレクトされた写真を加工して見せることになる訳である。

この時、セレクトされた写真の集合や加工された写真の集合といった、写真の集合が扱えると便利であり、幾つかの写真管理ソフト・現像ソフト等では、『アルバム』という機構が設けられていたりする。写真の集合の選び方は、人間による指定もありうるが、メタデータを用いて内包的に集合を定義する『スマートアルバム』(例えば、D700 で撮った写真の集合とか、2010年10月28日に撮った写真の集合とか。複数の条件を and/or でつないだ複合的な式を書くことも可能である) という機構を備えるものもある。

アルバムは写真の集合であり、写真の ID の集合により表現できる。また、アルバム中の写真の順序関係が必要であれば、集合ではなく、列としても良い。いずれにせよ、写真の ID のリストとして表現可能である。

2.4 ブラケットティング

多くのデジタル一眼レフカメラでは、シャッターボタンを押した時に、3枚とか5枚とかの写真の露出を変えて短時間に連続的に撮影する機能を持っている。これを『(露出) ブラケットティング』と呼ぶことにする。例えば、3枚単位になっている場合、1枚目は標準露出、2枚目は暗めに、3枚目は明るめに撮るといった具合である。² これは撮影時にきちんと露出を決めなくても、後で一番望ましい露出の写真を決めることができ便利であるし、複数の写真を合成した HDR 写真を作ることもできる。被写体の移動・変化やぶれを無視すれば、ブラケットティングして撮った3枚とか5枚の一連の写真は基本的に同じものを撮った写真の一部(同じ写真のバリエーション)といえ、関連付けて管理することが望ましい。しかしながら、ブラケットティングに関する情報の記録の仕方はメーカー(カメラ)毎に異なっているし、必ず設定した枚数全部連続して撮影されるとは限らないので、撮影時刻等の情報を併用したヒューリスティクスを用いないと違う写真を関連付けてしまうことになる。

2.5 加工された写真の参照

写真を WWW 上で利用する場合、オリジナルの画像データをそのまま表示するだけでなく、画面サイズや用途に応じてリサイズしたり、一部をトリミングしたり、あるいは、露出やホワイトバランス・γカーブを変えるなどの現像処理を行うなど、何らかの加工を行ったものを表示したいことが少なくない。こうした加工の仕方を決め打ちではなく、なるべく柔軟に WWW 上で設定可能にし、かつ、その結果を WWW 上で共有し、他の WWW サービスとの連携を容易にするため

² ホワイトバランスなど他のパラメーターを変えて撮る機能を持っているカメラもある。

には、各種設定や状態を REST アプローチに基づいて扱うのが望ましいといえる。[5] [9] このことを鑑みれば、ベースとなる（オリジナルの画像データ）の URI に対して、加工方法を表すパラメーターを付けた URI を用いて参照するのが良いといえる。

3 RAW 形式を扱うためのツール

3.1 ExifTool

ExifTool [3] は画像のメタデータを読み書きするためのソフトウェアで Perl のライブラリとコマンドライン・アプリケーションを備えている。ExifTool は JPEG/Exif のみならず、さまざまなメーカーのデジタルカメラの RAW 形式を含むさまざまな画像形式・メタデータをサポートしている。

ExifTool はさまざまな画像形式・メタデータをサポートしているため、データ形式や取りたい情報によってプログラムを使い分ける必要がなく、単純化できるというメリットもあるが、より大きなメリットは Exif タグで記録されていない情報を RAW 形式の写真データから得ることができることであろう。

例えば、被写体までの距離の情報は被写体の位置を知る上で有用な情報であるが、Exif タグに記録するカメラはほとんどなく、Exif タグだけでは事実上抽出不可能である。[7] しかしながら、RAW 形式には記録されている場合がある。例えば、Nikon D700 に AF 対応レンズを付けた状態で撮影した RAW 形式のデータの場合、FocusDistance タグにこの情報が記録されており、³ ExifTool を使って読みだすことができる。

3.2 dcraw

dcraw [1] はデジタルカメラの RAW 形式を解釈し標準的な画像形式に変換（このことを『現像』と呼ぶ）するためのソフトウェアである。dcraw は C 言語で書かれており、Linux やその他 UNIX 系 OS で動く他、Windows でも利用可能である。dcraw 自体がサポートする出力形式は PGM/PPM/PAM 形式と TIFF 形式だけ⁴であるが、ImageMagick 等の他の画像処理プログラムと併用することで JPEG をはじめとするさまざまな画像形式に変換することができる。

4 写真のセマンティクスをどう扱うか

4.1 『～の写真』

写真を探すという行為は、ある目的を満たす写真を探すということだといえる。例えば、『土偶の写真』とか『明治時代初期の京都市内の写真』とか『1930 年台のファッションの写真』というのはその一例である。こうした写真の検索にはしばしば写真に付けられたアノテーションやメタデータが用いられるが、これはある意味において、写真を属性の束からなる概念のようなものとして捉える行為だと看做することができるだろう。写真は、本来、『この写真』という風にしか指し得ないものかも知れないが、しばしば、『～の写真』という風に指されるような『概念上の写真』（の

³但し、これは近似値であり、その誤差はレンズ等に依存し、レンズによっては著しく不正確な場合もある。

⁴標準では色の解像度は 8bit であるが、16bit で出力することもできる。

インスタンスのようなもの)として扱われる。これは『写真のセマンティクス』という風に言い換えることができるかも知れない。

属性の束で示されるような『概念上の写真』はそれに属する無数の写真の集合である。前述の『スマートアルバム』は有限の写真の集合に対して、これを実現したものと看做することができる。多数の同様な写真の中から1枚(ないしは少数)の写真を選び出すという行為は、この概念上の写真を示すための代表を選ぶ行為だと看做することができる。これは符号化文字(抽象文字)に対して例示字形を付けることに似ているかも知れない。

『概念上の写真』は指定の詳細さによって、その集合の包含関係を考えることができる。例えば、『土偶の写真』の中には『青森県の土偶の写真』があるし、場所や時代をもっと特定することでより詳細に指定することもできる。こうした属性の束によって指されるものとしての『概念上の写真』の性質は、同様に視覚的に表現される記号である文字(特に、漢字)の場合と共通する部分が多いといえ、素性の集合(属性の束)によって文字を表現する Chaon モデル [6] と同様な手法で表現できるといえる。

しかしながら、『写真のセマンティクス』は文字の場合とは異なる部分もある。文字の場合、通常、作者はいない⁵『概念上の文字』はその文字を解する不特定多数によって共有されており、多少の揺れはあったとしても、『作者の意図』や『受け手の解釈』の介在する部分は少ない。言い換えれば、『文字のセマンティクス』はそれをやりとりする解釈共同体の中に置かれているといえる。しかしながら、写真の場合は写真を撮った人がいるし、写真に撮られた被写体が(少なくともその写真が撮られた時には)存在したはずである。そして、これらは属性の束としての写真、すなわち、『概念上の写真』という枠組に収まり切らないような面がある。

写真資料のデータベース(あるいは、そのための写真のセマンティクス)というものを考えた場合、写真が持つこの2つの側面をうまく扱う必要があると思われる。

4.2 どのようなシステムを構築すべきか

『概念上の写真』は素性の集合(および、それによって表される『写真』間の関係等)によって表されるようなものとして捉えることができ、CHISE-Wiki [10] を一般化したようなもの⁶によって扱うことができると考えられる。⁷

一方、『撮られたものとしての写真』というものを扱う場合、ひとつには、撮影された写真データ自体をどう指し示すかという問題として捉えることができるといえ、2節で議論したような、オリジナルの写真データやそれを加工したものをどう URI 等で参照するかというような問題として扱うことができるかも知れない。他方、『作者の意図』や被写体(被写体が人間であった場合、『モデルの意図』も考えられるし、同様に、『メイキャップアーティストの意図』や『コーディネーターの意図』なども考えられる。被写体が人工物であれば、その作者の意図も考えられる)に関する情報の幾つかは、撮影データや被写体に関するメタデータ等から読み取れるかも知れないし、無論、知り得ない(言語化できないような)ものもあるだろう。いずれにせよ、写真自体やそのメタデータなどで表現し得ない(解析し得ない)ようなものは検索できない訳であり、もし検索の対象としたいのであれば、原理的に暗黙知的なものとならざるを得ないようなものであったとしても、なん

⁵歴史的・伝説上の作者がいる場合はあるが、通常、誰かの著作物とは看做されないし、そのことが文字のセマンティクスを規定したりしない。

⁶現在、CHISE-Wiki を一般化して、文字以外の任意のオブジェクトを対象とした EGT というシステムを開発中である。

⁷この『概念上の写真』はキャラクター(コス)写真を対象にした場合、よりはっきりと意識されるかも知れない。例えば、Cure [2] の場合、ジャンルや作品、キャラクターとそのバリエーションといった階層的なカテゴリが付けられており、指定したカテゴリに属する写真を串刺し検索することができるようになっている。

らかの概念化・記号化を行う必要があるといえる。これは言い換えるならば、『概念上の写真』として扱わなければならないということである。

こうした問題を鑑みれば、結局、2節で議論したような写真データ自体のストレージとでもいうべきものと、素性の集合によって『概念上の写真』を表現する素性ストレージとでもいうべきものを組み合わせたようなシステムが望ましいのではないかと思われる。これはちょうど CHISE-Wiki と GlyphWiki [8] の関係に似たものという風に考えることができる。ただ、写真に対する視点の多重性は文字よりもはるかに多面的で複雑なものになり得るので、この点に配慮した設計を行うべきだろう。ただ、素性の集合という枠組は非常にプリミティブであるので、どのようなモデル化を行ったとしても、素性ストレージで扱えるようなものとなるのではないかと思われる。現実的には、Wiki 的な方法によって試行錯誤しながらモデルを改良していくということになるのではないかと思われ、素性ストレージのフロントエンドとなるユーザーインターフェースはこうしたモデル化の試行錯誤を支援するような機能を持つことが望ましいといえる。

5 おわりに

デジタル写真資料を WWW ベースで共有・整理・活用する場合の課題や取り得る手段について議論した。

一般的なデジタル一眼レフカメラに比べて、現在の標準的な配布フォーマットである JPEG や現在の一般的なパソコンの表示能力は、解像度・画素数、色空間や色深度といった点ではるかに限定されており、センサーが捉え得る情報の全てを画面上で表現することができない。しかしながら、こうしたことはハードウェアに関する原理的な問題というよりは、多分にソフトウェア (OS や GUI, WWW ブラウザー等) や画像フォーマットの問題であり、AdobeRGB 等の広色域色空間をサポートするディスプレイは比較的安価に入手できるようになっているし、デジタルテレビなどの家電の世界では 10bit の色深度をサポートする製品も普及しつつある。200dpi を越えるような高解像度のディスプレイが一般的でないのも、GUI や WWW ページが 100dpi 程度の解像度を想定して作られており、解像度独立になっていないことが一因であると考えられる。今話題となっている電子書籍も、iPad や Kindle 等のリーダの解像度がそれほど高くないため、紙に比べてはるかに低解像度の環境を想定して作られてしまう恐れがあるように思われる。しかしながら、人文情報系データベースの寿命や制作のためのコストを考えれば、より長いスパンで考えた方が良いのではないかと思われる。こうしたことを鑑みれば、現在利用可能な機材を最大限に活かしたデータ化を検討すべきであり、デジタル写真においてはマスターデータとして RAW 形式を基本とするのが望ましい。しかしながら、RAW 形式はベンダー・カメラ毎に形式が異なり、互いに互換性がなく、情報交換の点で難がある。しかしながら、この問題は、現像に関しては dcrw, メタデータに関しては ExifTool を用いることでほぼ解決するといえる。

(Wiki 的な手法を用いて) WWW 上で写真を扱う上で重要なことのひとつは Web API の設計であるが、別の WWW サービスからの簡単に利用するためには REST アプローチを採用するのが望ましいといえる。このことは、言い換えれば、写真に対してどのように URI を対応付けるかということであるが、基本的には撮影者と日付とオリジナルのファイル名の組合せを用いることで写真の ID や URI を作るができる。そして、撮影 (イベント) や『フィルムロール』といった単位で写真のマスターデータをシステムティックに管理することができる。

一方、写真のセマンティクスを考えた場合、こうした『撮影された写真』とは別に、素性 (メタデータやアノテーション等) の集合で指定されるような『概念上の写真』というものを考える必要

があるといえる。写真の整理や利用とは『撮影された写真』と『概念上の写真』を対応付ける作業に他ならず、また、写真の検索とは『概念上の写真』に対応する『撮影された写真』を捜し出す行為と看做すことができる。『概念上の写真』の持つ性質は (Chaon モデルにおける) 文字 (特に漢字) と同様な部分が少なからずあり、共通の枠組で扱えるのではないかと考えられる。しかしながら、写真にはこうした視覚的記号としての側面の他にも、『撮影した時に存在した被写体を写したもの』という側面や (作者がいて読者がいるような) 『作品』としての側面があり、こうしたことを鑑みれば、『概念上の写真』という視点だけでは明らかに不十分である。⁸しかしながら、コンピューターに載せようとする限り、形式化・機械可読化が必要であるのも事実である。よって、素性の集合やオブジェクト間の関係という枠組を用いて、こうした多面性や暗黙知的・感覚的なものの明示化・モデル化を行う必要があるといえる。このためには Wiki 的な手法の利用は有用ではないかと考えられる。

こうした観点に基づき、現在、『撮影された写真』のストレージ用のソフトウェア、および、CHISE-Wiki を一般化した素性ストレージである EgT というソフトウェアを開発中であり、この両者を統合した環境の実現を目指している。前者は現在の所、リサイズ機能と iPhoto からの変換用ユーティリティを備えるだけの簡単なものであるが、将来的には RAW 形式からの現像機能やトリミング等の簡単な加工機能と一括ダウンロード機能も追加する予定であり、ローカルとの同期機能の実現も検討している。

参考文献

- [1] Dave Coffin. Decoding raw digital photos in linux. <http://www.cybercom.net/~dcoffin/dcraw/>.
- [2] Cure. <http://ja.curecos.com/>.
- [3] Phil Harvey. Exiftool. <http://www.sno.phy.queensu.ca/~phil/exiftool/>.
- [4] ロラン バルト (Roland Barthes). 明るい部屋—写真についての覚書 (原題: La Chambre claire: Note sur la photographie). みすず書房, 1997 (原書: 1980) 年. 花輪 光 (訳).
- [5] Henry S. Thompson. Web Architecture and Naming for Knowledge Resources. In Takenobu Tokunaga and Antonio Ortega, editors, *Large-Scale Knowledge Resources*, Vol. 4938 of *LNAI*, pp. 334–343. Springer, 2008 年.
- [6] MORIOKA Tomohiko. CHISE: Character Processing Based on Character Ontology. In Takenobu Tokunaga and Antonio Ortega, editors, *Large-Scale Knowledge Resources*, Vol. 4938 of *LNAI*, pp. 148–162. Springer, 2008 年.
- [7] 田島孝治, 高田智和. 「景観文字調査」のための調査結果分類方法に関する一考察. 情処研報, Vol. 2010-CH-87, No. 6, pp. 1–6, 2010 年 7 月.
- [8] 上地宏一. GlyphWiki. <http://glyphwiki.org/wiki/GlyphWiki>.

⁸ロラン・バルトは「明るい部屋」[4]の中で、『ストゥディウム』と『プンクトウム』という概念を用いてこうした問題を指摘している。

- [9] 守岡知彦. データを生み出すデータのために. 人文科学とコンピュータシンポジウム論文集『サービス指向のデジタル技術へ～人文科学のポテンシャル～』, pp. 13-18. 情報処理学会, 2008年12月.
- [10] 守岡知彦. CHISE のセマンティック wiki 化の試み. 情処研報, Vol. 2010-CH-87, No. 8, pp. 1-8, 2010年7月.