Qi, Qunfen, Jiang, Xiang and Scott, Paul J.

Knowledge modeling for specifications and verification in areal surface texture

Original Citation

Qi, Qunfen, Jiang, Xiang and Scott, Paul J. (2012) Knowledge modeling for specifications and verification in areal surface texture. Precision Engineering, 36 (2). pp. 322-333. ISSN 0141-6359

This version is available at http://eprints.hud.ac.uk/14572/

Knowledge modeling for specifications and verification in areal surface texture

Qunfen Qi, Xiangqian Jiang* and Paul J. Scott

EPSRC Centre for Innovative Manufacturing in Advanced Metrology, University of Huddersfield, Huddersfield HD1 3DH, UK

*Corresponding author: Professor Xiangqian Jiang. E-mail: x.jiang@hud.ac.uk, telephone: +44-1484 473634, fax: +44-1484 472161

Abstract: The 25178 series of standards in areal surface texture covers terms and definitions for specification and verification operators and is being developed by work group (WG) 16 in the International Standards Organization (ISO) TC 213. As there are many innovative concepts and definitions included in these standards, it is often considered difficult for mechanical engineers to comprehend and for computing engineers to apply in computing science. This paper presents the utilization of category theory to model sophisticated knowledge in the field of areal surface texture. The ISO 25178 series can be divided into specification and verification series according to the principles of Geometrical Product Specifications (GPS). In the category model, categories and objects are used to represent different knowledge structures; arrows and pullbacks are used to sketch diverse connection between objects; functors are utilized to reveal the structure-preserving mapping between categories in specification and verification. In this paper the function of pullbacks is considered to be a pullback inference mechanism since most of the objects in the model can be determined by different pullbacks. The knowledge model in this paper is the foundation for developing a design and measurement information system in areal surface texture for manufacturing industry.

Keywords: areal surface texture, knowledge modeling, category theory, specification, verification

1. INTRODUCTION

With the advance in surfaces assessment, it was found that some of the surface profile parameters (such as Ra and Rz) had very limited value in relating the surface to its functional effectiveness. Had instrument development, in relation to data acquisition and signal processing, proceeded in advance of the subject of surface characterization, the probable development and specification of parameters would have been more logical through areal data collection analysis [1]. It shows that areal surface texture analysis is now essential wherever a complete assessment of the surface is required to enable the selection of the most appropriate surface texture to achieve a required functionality. Conscious of the "parameter rash" [2], the research group of Prof. Stout developed a primary set of areal parameters named "Birmingham 14" parameters [3] in 1993. Later, the European project "SURFSTAND" [4] under the leadership of Huddersfield University improved these parameters by working on the correlation with functional specifications, and prepared the basis for ISO 25178-2 [5] of which the first draft was developed in April 2006. Currently, the ISO 25178 series of areal surface texture standards concerning terms and definitions, specifications and verification operators is being developed by WG 16 in TC 213. It is the first and foremost series of standard providing a redefinition of the foundations of surface texture, and based upon the principle that nature is intrinsically 3D. It is anticipated that future work will extend these new concepts into the domain of 2D profile metric surface analysis, requiring a total revision of all current surface texture standards (ISO 1302, ISO 4287, ISO 4288, ISO 115652, ISO 12085, ISO 13565 series, etc). Many innovative concepts are introduced in the ISO 25178 series of documents. Table 1 shows all areal surface texture standards in the general GPS matrix [6]. Heretofore, ISO 25178 part 1 [7] defines the indication of areal surface texture as shown in figure 1; part 2 defines the terms, definitions and surface texture parameters which include field and feature parameters [8]; part 3 [9] defines areal surface texture specifications operators; part 6 series [10-15] define the measurement methods and instruments; part 7 series [16-19] define calibration requirements and software measurement standards. Here, parts 1-3 define the requirements for specifications and parts 6-7 described the characteristics for verification.

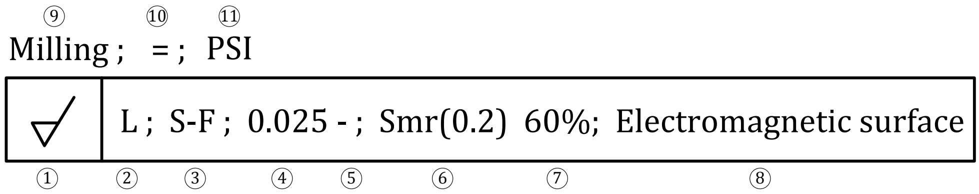
In 2010, ISO 25178-6, ISO 25178-601, ISO 25178-602 and ISO 25178-701 became the first four published standards in areal surface texture. According to the schedule of WG16, other standards will be published shortly. Areal surface texture characterization in manufacturing industry will be more widely used. As there are many innovative concepts and definitions involved in this series, it is often considered difficult for mechanical engineers to comprehend and for computing engineers to apply in computing science. Moreover, the level of understanding designers have for specifications knowledge of areal surface texture is still unsatisfactory; and there is no effective reference for metrologists to arrange a series of measurement processes for areal surface texture.

Chain link No.	**Geometrical characteristic of feature**	**Areal surface texture standards**
1	Product documentation indication - Codification	ISO 25178-1(D)
2	Definition of tolerances – Theoretical definition and values	ISO 25178-2(D)
3	Definition for actual feature – Characteristic or parameter	ISO 25178-3(D)
4	Assessment of the deviations of the workpiece – Comparison with tolerance limits	
5	Measurement equipment requirements	ISO 25178-6, 25178-601, 25178-602, 25178-603(D), 25178-604(D), 25178-605(D)
6	Calibration requirements – Measurements standards	ISO 25178-70(D), 25178-71(D), 25178-701, 25178-702(D)

Note: The symbol (D) is standards under development

Table 1 Areal surface texture standards in general GPS matrix

The aim is to express specifications and verification knowledge involved in areal surface texture, and help designers and metrologists to utilize areal surface texture characterization effectively. This paper utilizes a graphical category model which is based on category theory to structure the knowledge. The specifications model can generate a complete series of areal surface texture specifications for designers. According to the specified specifications, the verification model can produce series related verification information to guide the measurement procedure and measurement result treatment for metrologists. The knowledge model in this paper is the basis for developing a design and measurement information system in areal surface texture for manufacturing industry.



① Surface texture graphical symbol
② Type of tolerance upper (U) or lower (L)
③ Type of scale-limited surface
④ Nesting index – S filter
⑤ Nesting index – F operator or L filter
⑥ Areal parameter
⑦ Limit value
⑧ Other non-default(s)
⑨ Manufacturing process
⑩ Surface texture lay
⑪ Other informtion

Figure 1. Control elements in indication of areal surface texture requirements on engineering drawings

2. CATEGORY THEORY APPLIDED IN AREAL SURFACE TEXTURE

2.1 Category theory

Category theory is a branch of mathematics that has been developed over the last 60 years since it has been found that many properties of mathematical system can be unified and simplified by a presentation with diagrams of arrows. It explores the relationships between different kinds of mathematical objects, and ignores unnecessary detail to give general definitions and results. It is a high-level (abstract) and efficacious language that focuses on how things behave rather than on what their internal details are [20-21]. There are three important concepts in category theory which are often used when utilizing it in areal surface texture – categories, pullbacks and functors.

A **category C** consists of a collection of objects A, B, C, … and a collection of morphisms or arrows which are the abstraction derived from structure-preserving mappings between objects f: $A \rightarrow B$, g: $B \rightarrow C$,…, that are closed under composition and satisfy the following conditions.

- For each arrow f there are given objects: dom(f), cod(f) called the domain and codomain of f. We write: f: $A \rightarrow B$ or $A \xrightarrow{f} B$ to indicate that A = dom(f) and B= cod(f).
- Given arrows f: $A \rightarrow B$ and g: $B \rightarrow C$, that is, with: cod(f) = dom(g), there is given an arrow: $g \circ f$: $A \rightarrow C$, called the composite of f and g.
- For each object A, there is an identity arrow id_A: $A \rightarrow A$ satisfying the identity law: for any arrow f: $A \rightarrow B$, $id_B \circ f = f$ and $f \circ id_A = f$.

The collection of all morphisms from A to B in category **C** is denoted $\hom_{\mathbf{C}}(A,B)$ and called the **hom-set** between A and B (the collection of morphisms is not required to be a set). A number of types of morphisms are defined in category theory are monic (monomorphism), epic (epimorphism) and isomorphic. In the category **Set** (objects are sets, morphisms are functions), monic is same as injection (one-to-one function), epic is same as surjection (onto) and isomorphic is same as bijection (one-to-one and onto). Note that a morphism may not be an isomorphism even it is monic and epic.

A **pullback** of the pair of arrows f, g with cod(f) = cod (g) as shown in figure 2.a is an object P and a pair of arrows p_1 and p_2 as shown in figure 2.b such that $f \circ p_1 = g \circ p_2$. And if z_1: $Z \rightarrow A$ and z_2: $Z \rightarrow B$ are such that $f \circ z_1 = g \circ z_2$, then there exists a unique u: $Z \rightarrow P$ with $z_1 = p_1 \circ u$ and $z_2 = p_2 \circ u$. The related picture is shown in figure 2.c. A product of two objects A and B is an object $A \times B$ together with two projection arrows π_1: $A \times B \rightarrow A$ and π_2: $A \times B \rightarrow B$. Thus, object $A \times B$ and arrows π_1 and π_2 is the pullback of C, and arrows f, g. Consider the diagram in figure 2.f which e is an equalizer of $f \circ \pi_1$ and

$g \circ \pi_2$ and $p_1=\pi_1 \circ e$, $p_2=\pi_2 \circ e$. Then E, p_1, p_2 is a pullback of C, f and g.

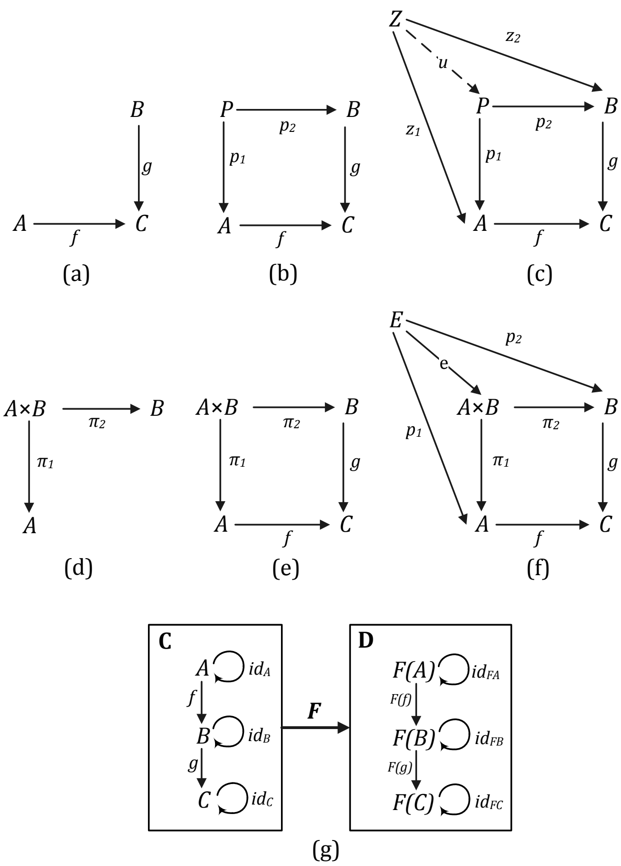


Figure 2. Arrows, pullbacks and functor

An arrow between categories is termed a functor if it satisfies some structure-preserving requirements:

(1) For each arrow f: $A \rightarrow B$ in **C**, there is an arrow $F(f)$: $F(A) \rightarrow F(B)$ in **D**.

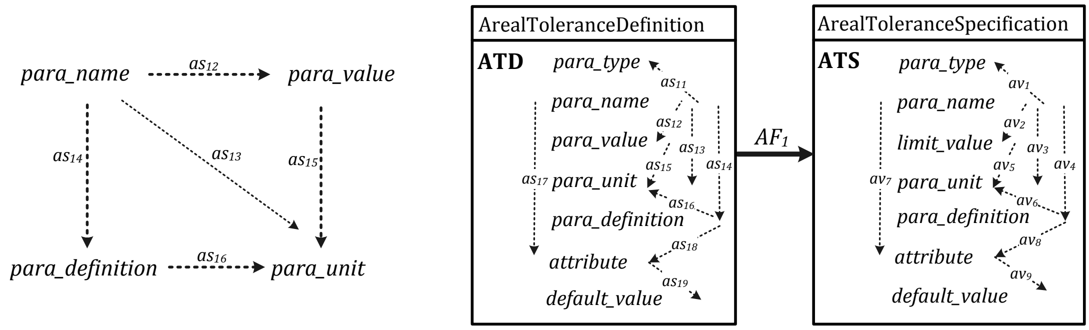
(2) For each object A in **C**, the equation $F(id_A)=id_{FA}$ holds in **D**.

(3) For each pair of arrows $A \xrightarrow{f} B \xrightarrow{g} C$ in **C**, the equation $F(g \circ f)=F(g) \circ F(f)$ holds in **D**.

This type of arrow provides the facility for transforming from one category type to another category type. **Functors** are therefore basically structure-preserving morphisms from a source category to a target category. An obvious case is when the shape of the target category is determined by the functor, that is it accommodates all assignments from the source category and has no other structure of its own. However, one of the major features of functors is that it connects two different structures by structure-preserving mapping. One particular example is a forgetful functor which is defined from a category of algebraic gadgets (group, modules, vector spaces, etc) to the category of sets. The forgetful functor leaves the objects and the arrows as they are, remembering only the underlying set and regardless of their algebraic properties. Furthermore, functors can also be monic so that the target category contains equal or more structure than the source category. The functor from a subcategory onto the category on which it is founded is an example of such morphism.

2.2 Category model for areal surface texture

The knowledge about areal surface texture includes massive diverse concepts and structures which cover specification definitions, definition categories, semantic understanding, algebraic structures, structured entities and relationships between all of them. The diversification of the knowledge makes it hard to apply in computing science. Based on characteristics of category theory, it can use categories to express all of the different kinds of structures in areal surface texture, and objects and arrows in a category to describe different elements in structures and relationships between elements respectively. The relationships between different structures (categories) can be expressed as functors. Hence, category theory ignores the unnecessary details of different definitions and structures and focuses on the categories and relationships between and in them. The convenience of category theory to describe complex relationships between different definitions was used for structured entities in profile surface texture [22-23] and cylindricity [24]. Areal surface texture has never been structured before. In this paper, the category model which is based on category theory is applied to model the definitions, structures and relationships between them in areal surface texture.



(a) (b)

Figure 3. An example of category model for areal surface texture

Figure 3 gives an example of how to represent the tolerance definition in areal surface texture. **ATD** is a category which represents the tolerance definition of areal surface texture. It is composed of seven objects (*para_type*, *para_name*, *para_value*, *para_unit*, *para_definition*, *attribute*, *default_value*) and nine arrows (as_{11}, as_{12}, as_{13}, as_{14}, as_{15}, as_{16}, as_{17}, as_{18} and as_{19}). The arrow as_{11} states the collection of morphisms from *para_name* to *para_type* written as $\hom_{\mathbf{ATD}}$(*para_name*, *para_type*) which is epic. It states every parameter belongs to a kind of parameter type, for example the parameter *Str* (texture aspect ratio) is classified by spatial parameters. The arrow as_{12} as $\hom_{\mathbf{ATD}}$(*para_name*, *para_value*) is epic which representing the parameter value is decided by the parameter name. For instance, for a specified honing surface, the parameter value of parameter *Sal* (auto-correlation length) can be 0.06mm, and parameter *Sa* of 0.728μm. The arrow as_{13} as $\hom_{\mathbf{ATD}}$(*para_name*, *para_unit*) is epic which shows that every parameter has a related unit. The arrow as_{14} as $\hom_{\mathbf{ATD}}$(*para_name*,*para_definition*) is isomorphism which express that every parameter has a unique parameter definition. The arrow as_{15} as $\hom_{\mathbf{ATD}}$(*para_value, para_unit*) is epic which denotes that every parameter value should include a unit. The arrow as_{16} as $\hom_{\mathbf{ATD}}$(*para_definition*, *para_unit*) is epic which indicates that the parameter definition determines the type of parameter unit. The arrow as_{17} as $\hom_{\mathbf{ATD}}$(*para_name*, *attribute*) is epic which means some parameters have an attribute. For instance, the attribute of parameter *Str* is the fastest/slowest decays to *s* (with $0 \leq s < 1$).

The arrow as_{18} as hom_{ATD}(*para_defintion*, *attribute*) is epic which presents that it is the definition of parameter which determines the attribute. The arrow as_{19} as hom_{ATD}(*attribute*, *default_value*) denotes that every attribute has a default value (1:N relationship). For example, the default value of *s* which is the attribute of parameter *Str* is 0.2. Data examples for characteristic of areal surface texture parameters are shown in Table 2.

Parameter type	Parameter	Default unit	Attribute	Default value
Height parameters	*Sq*	µm	-	-
	Ssk	Unitless	-	-
	Sa	µm	-	-
Spatial parameters	*Sal*	µm	fastest decay to a specified values s, with 0≤ s ≤1	*s*=0.2
	Str	Unitless	fastest & slowest decay to s, with 0≤ s ≤1	*s*=0.2
Functions and related parameters	*Vvv*	ml/m^2	material ratio *p*	*p*=80%
	Vvc	ml/m^2	material ratios *p* and *q*	*p*=10%, *q*=80%
	Vmp	ml/m^2	material ratio *p*	*p*=10%
	Vmc	ml/m^2	material ratio *p* and *q*	*p*=10%, *q*=80%
	Sxp	µm	material ratio *p* and *q*	*p*=2.5%, *q*=50%
Feature parameters	*Spd*	$1/mm^2$	Wolfprune Nesting Index X%	X%=5%
	Spc	1/mm	Wolfprune Nesting Index X%	X%=5%
	S5p	µm	Wolfprune Nesting Index X%	X%=5%
	S5v	µm	Wolfprune Nesting Index X%	X%=5%

Table 2 Data examples for characteristic of areal surface texture parameters [9]

According to the concept of pullbacks, the structure as shown in figure 3.a is a pullback. Here, (*para_name*, as_{12}, as_{14}) is the pullback of (*para_unit*, as_{15}, as_{16}) as $as_{15} \circ as_{12} = as_{16} \circ as_{14}$. In figure 3.b, AF_1:**ATD→ATS** is the functor between categories **ATD** and **ATS**. In this paper, **ATD** is one of categories in specification and **ATS** is one of categories in verification. Thus, functor AF_1 is one of mappings between specification and verification. According to the definition of functors, for each object and arrow in category **ATD**, there is a mapped object and arrow in category **ATS**. Therefore, for **ATD**-objects *para_value* and *para_name*, there are AF_1 (*para_value*), and AF_1 (*para_name*) in **ATS**-

objects, and AF_1 (*para_value*) = *limit_value*, AF_1 (*para_name*) = *para_name* in **ATS**-objects. Similarly, for **ATD**-arrows as_{11} and as_{12}, there are $AF_1(as_{11})$, and $AF_1(as_{12})$ in **ATS**-arrows, and $AF_1(as_{11})=av_1$, $AF_1(as_{12})=av_2$. The functor AF_1 here is a covariant functor which is preserves the directions of arrows, i.e., every arrow $as_i:A\rightarrow B$ is mapped to an arrow $F(as_i)$: $F(A) \rightarrow F(B)$. Here, the **ATD**-objects in specification and **ATS**-objects in verification are independent, and they are however related by the so called "Duality Principle" [25] in GPS. For example, the object *para_value* in **ATD** is the limit value for the assigned parameter in specification, the object *limit_value* in **ATS** will be the same limit value when the specification is interpreted to verification.

3. KNOWLEDGE MODELING FOR SPECIFICATION OF AREAL SURFACE TEXTURE

Currently, more and more academic areas and industries are beginning to apply areal surface texture measurement to investigate the quality and function relationships of surface. However, no applications for areal surface texture specifications exist in manufacture design so far. As the areal surface texture standards series will be published in the near future, it is important to provide designers with an unambiguous areal surface texture specification process model where there are high accuracy requirements for the surface.

3.1 The specification process of areal surface texture

Considering all of the published and unpublished standards in areal surface texture, the specification process of areal surface texture is modeled as shown in figure 4. Desired functions and other information such as manufacturing process and surface materials should be the inputs for a function design of areal surface texture. Different surface components or parts may have different inputs options. All of the specification control elements defined in ISO 25178-1 (see figure 1) should be established according to the inputs and the inference of relationships. After the inference procedure, all of the inferred control elements such as parameter limit value, filtration, nesting index and other related information can be combined into a complete areal surface texture specification. Then the specification can be generated by a CAD system to an indication as an engineering drawing and saved to specifications data.

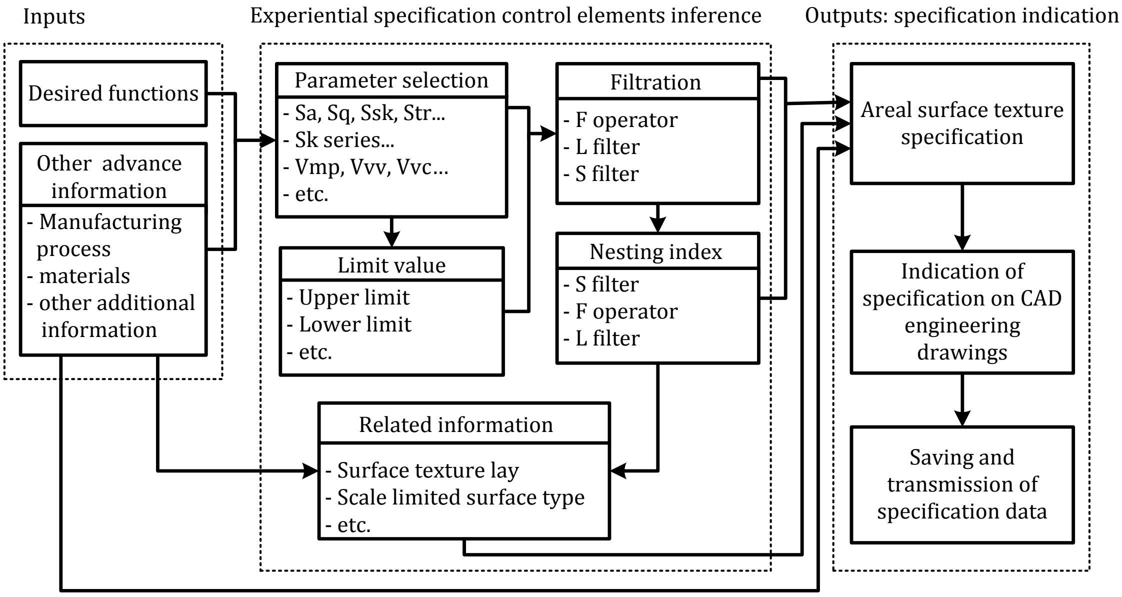


Figure 4. The specification process of areal surface texture

3.2 Category model for specifications of areal surface texture

According to the category model, figure 5 is the input category **AI** in specifications. **AI**-objects are the elements which the designers need to input for designing the specification. The arrow as_1 as hom_{AI}(*surface_function*, *material*) states the function of the surface is one of the determining factors for characteristic of material (1:N relationship). The arrow as_2 as hom_{AI}(*material*, *manufacturing_process*) is epic which represents different types of materials having related appropriate manufacturing processes. For instance, for a surface with optical material, related manufacturing processes can be grinding or polishing etc. The arrow as_3 as hom_{AI}(*surface_function*, *manufacuring_process*) shows that the function of the surface is one of the determining factors for a manufacturing process (1:N relationship).

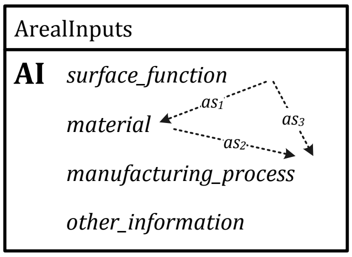


Figure 5. The input category for areal surface texture

AC-objects are the eleven control elements in indication of areal surface texture requirements on engineering drawings as shown in figure 1. Category **AC** is the most important part for an areal surface texture specification, and is inherited by three different categories **ACO**, **ATD** and **AFC** which belong to the first three chain links respectively in the general GPS matrix (see table 1). Here, AI_j denote the inherited relationships between categories. **ACO-**objects are the two elements related to specification indication. Object *indication_type* illustrates graphical symbols for three different manufacturing process types (as shown in figure 7); object *specification_type* presents upper and lower specification limit U or L. Category **ATD** is described in section 2.2.

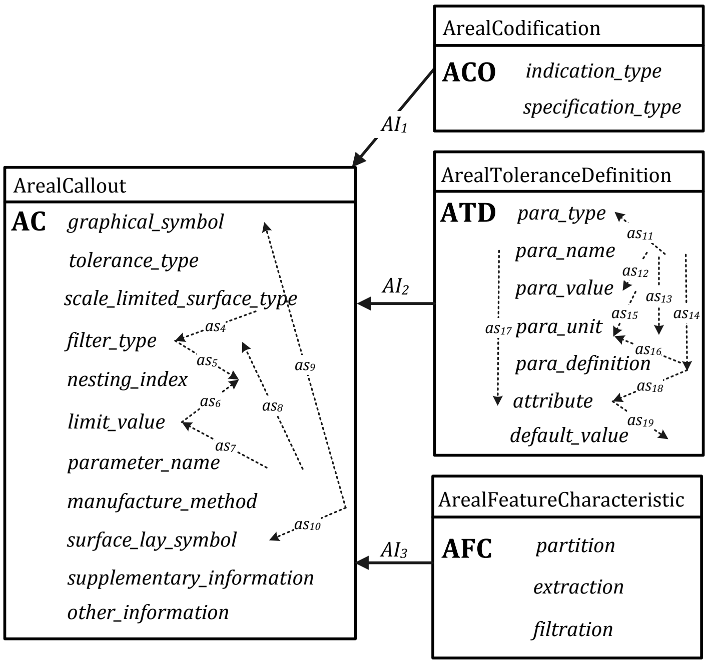


Figure 6. The input category for areal surface texture

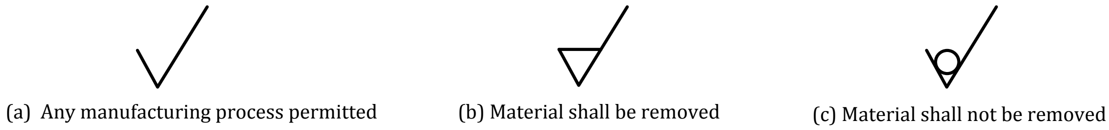


Figure 7. Three indication types for areal surface texture specification

Category **AFC** represents the feature characteristic in areal surface texture. It is composed of partition, extraction and filtration which are the three feature operations in GPS [25]. It is inherited from these three categories **AP**, **AE** and **AF** respectively, and category **ANI** is inherited from **AF** as shown in figure 8.

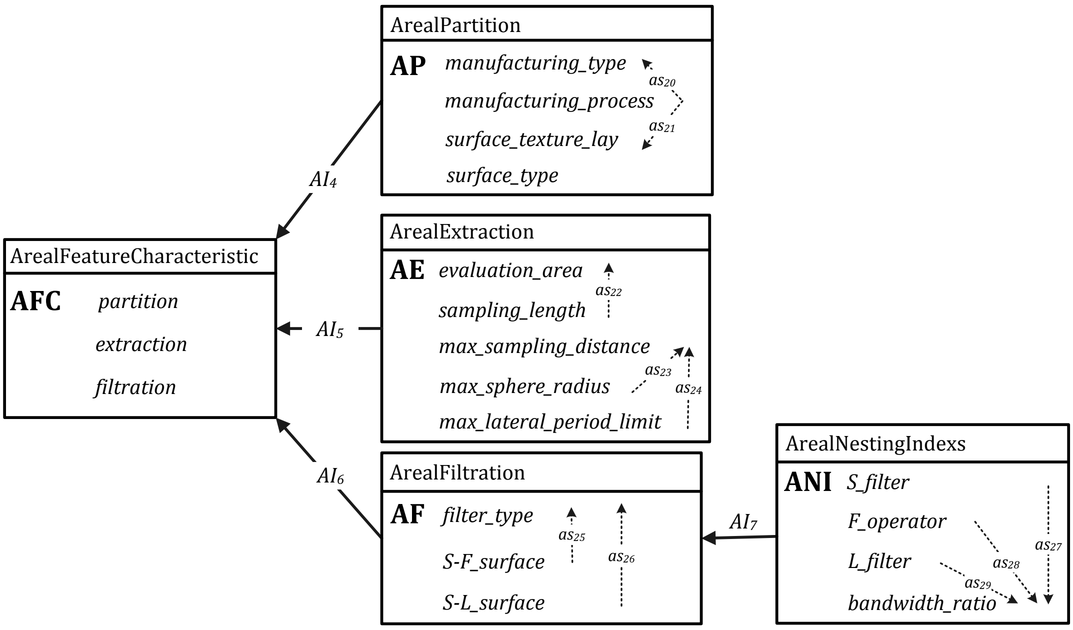


Figure 8. Category **AFC** and the inherited categories

Category **AP** represents the partition operation in specification. There are four objects in this category. The arrow as_{20} as $hom_{\mathbf{AP}}$(*manufacturing_process*, *manufacturing_type*) is epic which states that every manufacturing process belongs to a kind of manufacturing type such as "material shall be removed" type or "material shall not be removed" type. The arrow as_{21} as $hom_{\mathbf{AP}}$(*manufacturing_process*, *surface_texture_lay*) means every manufacturing process will generate different indication types of surface lay such as "=", "X" and "C" [26](1:N relationship). The

AP-object *manufacturing_process* and **AI**-object *manufacturing_process* are independent in each category although they refer to the same content. The relationship between these two objects is presented by pullback AP_1. Moreover, the arrows related with *manufacturing_process* in each category are also independent and are not related in any sense.

Category **AE** represents the extraction operation in specification. Five objects are involved. The arrow as_{22} as $\text{hom}_{\mathbf{AE}}$(*sampling_length*, *evaluation_area*) is isomorphism which expresses that evaluation area can be calculated according to the sampling length. The arrow as_{23} as $\text{hom}_{\mathbf{AE}}$(*max_sphere_radius*, *max_sampling_distance*) is isomorphism which means that the value of max sphere radius determines the value of max sampling distance for mechanical surfaces. The arrow as_{24} as $\text{hom}_{\mathbf{AE}}$(*max_lateral_period_limit*, *max_sampling_distance*) is isomorphism which means that the value of max lateral period limit decides the value of max sampling distance for optical surfaces.

There are three **AF**-objects involved in the filtration operation in specification. The arrow as_{25} as $\text{hom}_{\mathbf{AF}}$(*S-F_surface*, *filter_type*) expresses that S-F surface has a related filter type which includes S filter and F operator (1:N relationship). The arrow as_{26} as $\text{hom}_{\mathbf{AF}}$(*S-L_surface*, *filter_type*) expresses that S-L surface has a related filter type which includes S filter and L filter (1:N relationship).

Category **ANI** is inherited from Category **AF**. Four **ANI**-objects present the nesting index for different filters. The arrow as_{27}, as_{28} and as_{29} means the ratio between nesting index for S filter and F operator, or S filter and L filter are the bandwidth ratio.

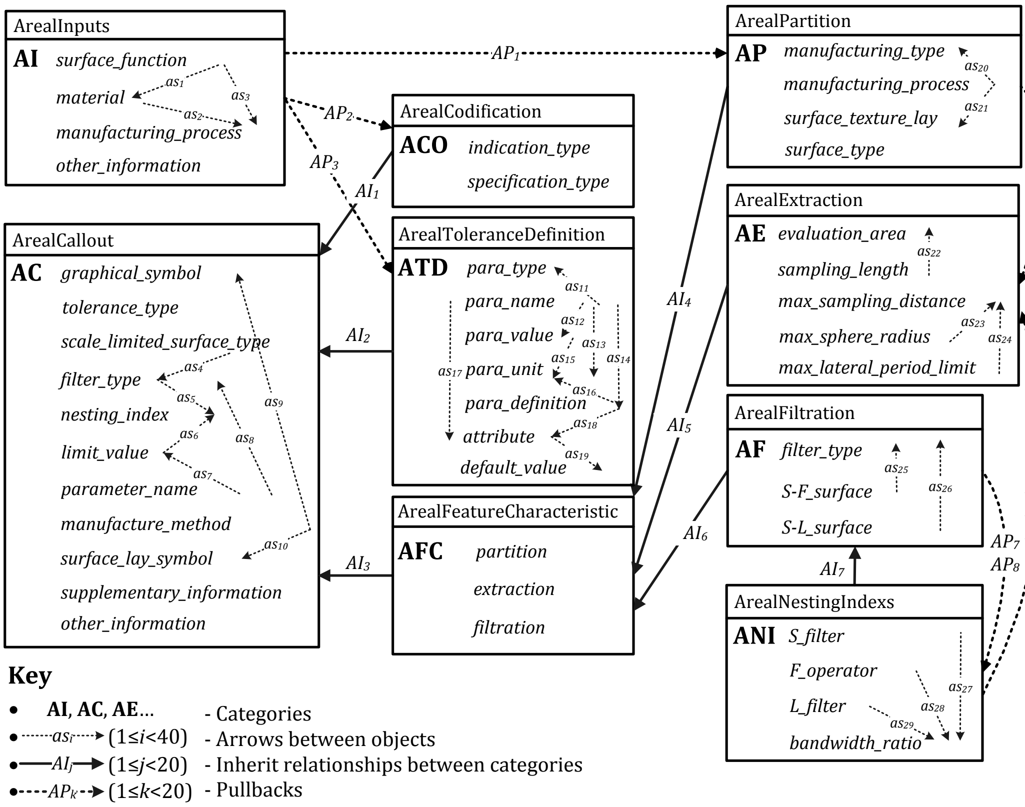


Figure 9. The category model for areal surface texture specifications (high-level abstract diagram)

According to the categories structures stated above, the whole high-level abstract category model for specifications of areal surface texture is showing in figure 9. Here, dashed arrows (AP_k) indicate pullbacks between different objects. The relationships between objects in different categories are expressed by pullbacks as described in section 2. The list of all the pullbacks in the specification model is shown below:

- AP_1 (*determine*: *manufacturing_process*) := **AI**-object: *manufacturing_process* → **AP**-object: *manufacturing_process*;
- AP_2 (*determine*: *indication_type*) := **AI**-object: *manufacturing_process* → **ACO**-object: *indication_type*;
- AP_3 (*determine*: *para_name* × *para_value*) := **AI**-objects: *functional_surface* × *material* × *other_information* → **ATD**-objects: *para_name* × *para_value*;
- AP_4 (*determine*: *max_sampling_distance* × *max_sphere_radius*) := **AP**-object: *surface_type* × **ANI**-object: *S_filter* → **AE**-objects: *max_sampling_distance* × *max_sphere_radius*;

- AP_5 (*determine*: *max_sampling_distance* × *max_lateral_period_limit*) := **AP**-object: *surface_type* × **ANI**-object:*S_filter* → **AE**-objects: *max_sampling_distance* × *max_lateral_period_limit*;
- AP_6 (*determine*: *evaluation_area* × *sampling_length*) := **ANI**-objects: *F_operator* × *L_filter* → **AE**-objects: *evaluation_area* × *sampling_length*;
- AP_7 (*determine*: *S_filter* × *L_filter*) := **AF**-object: *S-L_surface* → **ANI**-objects: *S_filter* × *L_filter*;
- AP_8 (*determine*: *S_filter* × *F_operator*) := **AF**-object: *S-F_surface* → **ANI**-objects: *S_filter* × *F_operator*.

Figure 10 gives an example of pullback structure AP_4 - the deduction of **AE**-objects *max_sampling_distance* and *max_sphere_radius*. The product of object *surface_type* in category **AP** and object *S_filter* in category **ANI** determines **AE**-objects *max_sampling_distance* and *max_sphere_radius*. In the pullback structure, the objects *surface_type* and *S_filter* from the product of categories **AP** and **ANI** constitute a subcategory **SAA**. Since $\pi_1 p_4 \circ \lambda_1 p_4 = \pi_2 p_4 \circ \lambda_2 p_4$, (**SAA**×**AE**, $\pi_1 p_4$, $\pi_2 p_4$) is the pullback of (AP_4 (…), $\lambda_1 p_4$, $\lambda_2 p_4$). Here, AP_4 (…) is a category with only one object and one identity arrow. Data examples of AP_4 are shown in Table 3. For example, if the nesting index of S filter is 0.1 µm for a mechanical surface, the max sampling distance and max sphere radius are 0.02 µm and 0.07µm respectively when a stylus instrument is applied. For an optical surface with the same S filter, they are 0.03 and 0.1 µm respectively.

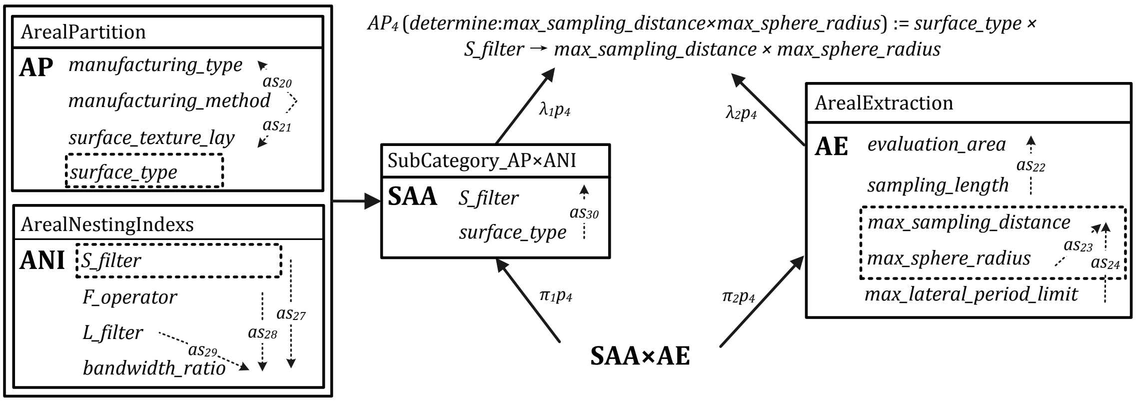


Figure 10. An example of pullback AP_4 – the determination process of **AE**-objects

max_smapling_distance and *max_sphere_radius*

AP	ANI	AE	
surface_type	*S_filter* (μm)	*max_sampling_distance*(μm)	*max_sphere_radius*(μm)
Mechanical surface	0.1	0.02	0.07
Optical surface	0.1	0.03	0.1
Mechanical surface	2.5	0.5	2
Optical surface	2.5	0.8	2.5

Table 3 Data examples of pullback AP_4

According to the pullbacks between objects in different categories, most of the objects in the model can be determined. Then the objects in **AC** can be inferred by this pullback inference mechanism. Then the specification can be established and the indications of it can be generated to show in engineering drawings.

4. KNOWLEDGE MODELING FOR VERIFICATION OF AREAL SURFACE TEXTURE

According to a specified specification, the metrologists measure the areal surface texture and determine whether the surface is qualified or not in the manufacturing step. This is the verification process. Figure 11 shows the verification process model for areal surface texture. There are three steps to obtain the final measurement results. In the “measurement preparation” step, metrologist analyzes the specification, and translates it to measurement specification which will be used to generate an appropriate measurement strategy with the considering of measurement conditions. Following the measurement strategy, metrologist carries out the measurement and obtains the measurement data. In this step, the metrologist selects the different options in the form removal and filtration parts. According to the data treatment selection, the software calculates the numerical result of the specified parameter in the last step. Based on the numerical result and uncertainty estimation, the metrologist should provide conformance or non-conformance with the specified specification. Finally, the measurement result will be feedback to the design stage in order to compare with the desired function which will help improve functional design.

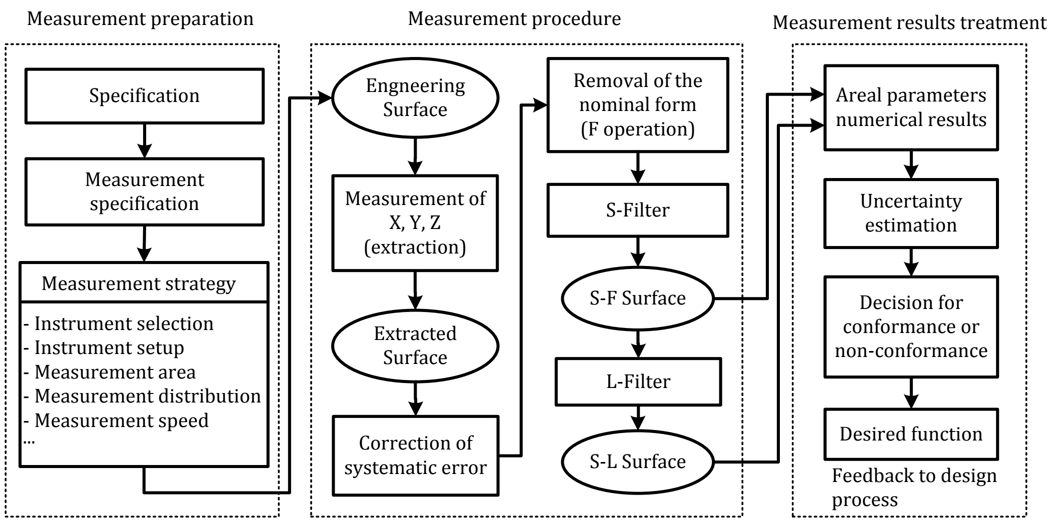


Figure 11. The verification process of areal surface texture

Figure 12 is part of the category model for the verification of areal surface texture. Here, the arrows in the same category are denoted by av_i in order to differentiate them from arrows as_i in specification. Category **AMS** is mapped from the specification category model. It includes four objects (*tolerance_specification*, *partition*, *extraction* and *filtration*) which are inherited by five categories **ATS**, **APV**, **AEV**, **AFV**, **ANIV** respectively. These five categories are mapped from the categories (**ATD**, **AP**, **AE**, **AF**, **ANI**) in specification, written as AF_1: **ATD** → **ATS**, AF_2: **AP** → **APV**, AF_3: **AE** → **AEV**, AF_4: **AF** → **AFV**, AF_5: **ANI** → **ANIV**. Following the explanation of the functor AF_1 which is described in section 2.2, every object and arrow in the category is mapped to the objects and arrows in another category, so are the pullbacks between different objects such as $AP_4 \rightarrow AP_{17}$, $AP_5 \rightarrow AP_{18}$, $AP_6 \rightarrow AP_{19}$, $AP_7 \rightarrow AP_{20}$, $AP_8 \rightarrow AP_{21}$.

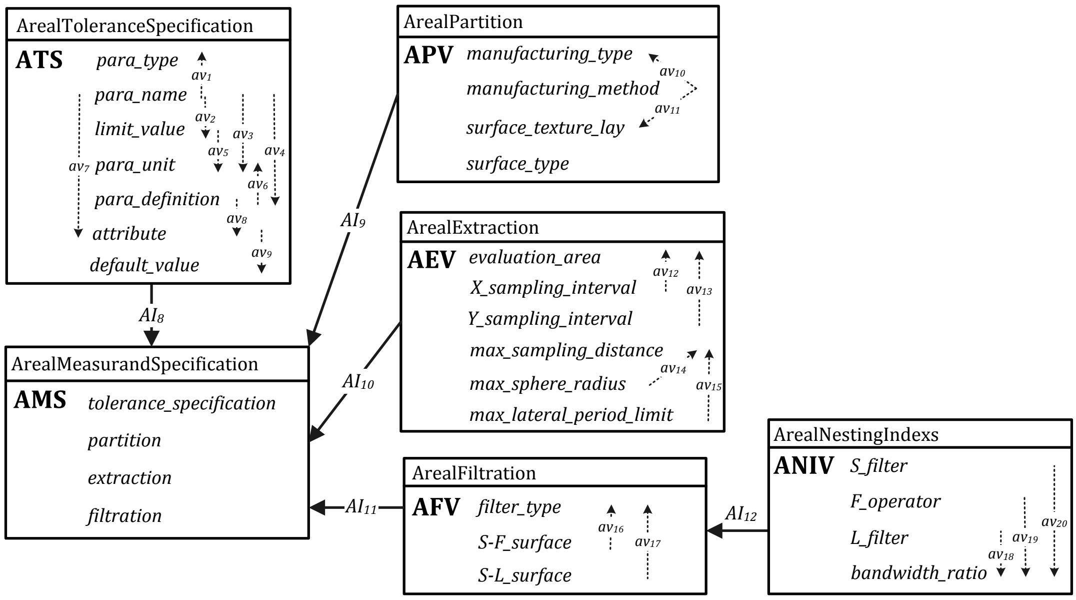


Figure 12. Category **AMS** and the inherited categories **ATS**, **APV**, **AEV**, **AFV** and **ANIV**

Figure 13 is the category **AME** in verification of areal surface texture. Seven **AME**-objects are the elements presenting characteristic of measurement equipment. The arrows av_{21}- av_{26} mean that the type of instrument determines all the instrument characteristic such as the repeatability of the instrument, the measure range, lateral and vertical resolution, the software functions and installation conditions etc.

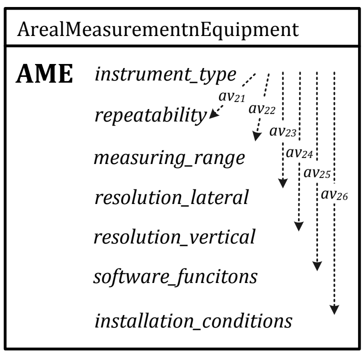


Figure 13. Category **AME**

Category **ACR** demonstrates the calibration requirements in the verification process. Six **ACR-**objects are required to characterize instrument calibration. The arrows av_{27} and av_{28} mean all kinds of measurement standards have related assessed parameters and measurement methods; the arrows av_{29} - av_{33} state that all the characteristics in calibration operation should be considered in the estimation process of measurement uncertainty. The arrow av_{34} means that every assessed parameter has its result.

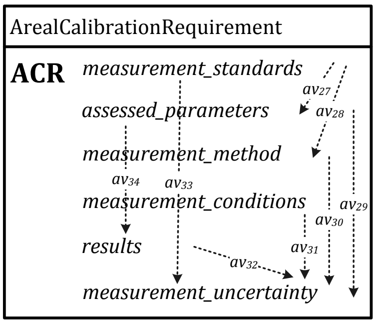


Figure 14. Category **ACR**

Category **AMR** presents the measurement result in the verification process. **AMR-**object *uncertainty_range* states the estimated range of measurement uncertainty in the verification process; object *accept_or_reject* denotes the measurement results whether the surface is accepted, rejected or in the uncertainty range. The arrow av_{35} means the uncertainty range of verification will contribute to structuring the conformance and non-conformance zone which will be used to determine the measurement result.

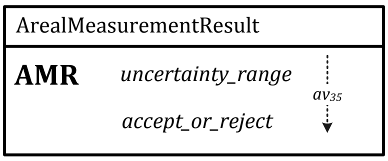


Figure 15. Category **AMR**

Figure 16 is the whole high-level abstract category model for verification of areal surface texture. By the pullback inference mechanism, pullbacks AP_k in verification can determine most of the objects in different categories. The details of every pullback in verification are shown as follows:

- AP_9 (*determine*: *resolution_lateral* × *resolution_vertical*) := **ATS-**objects: *para_name* × *limit_value* → **AME-**objects: *resolution_lateral* × *resolution_vertical*;
- AP_{10} (*determine*: *software_functions*) := **ATS-**objects: *para_name* × *limit_value* → **AME-**object: *software_functions*;
- AP_{11} (*determine*: *measurement_standards* × *assessed_parameters*) := **ATS-**object: *para_type* × **AME-**object: *instrument_type* → **ACR-**objects: *measurement_standards* × *assessed_parameters*;
- AP_{12} (*determine*: *instrument_type*) := **APV-**object: *surface_type* → **AME-**object: *instrument_type*;
- AP_{13} (*determine*: *measuring_range*) := **AEV-**object: *evaluation_area* → **AME-**object: *measuring_range*;
- AP_{14}(*determine*: *resolution_lateral* × *resolution_vertical*) := **AEV-**objects: *X_sampling_interval* × *Y_sampling_interval* → **AME-**objects: *resolution_lateral* × *resolution_vertical*;
- AP_{15} (*determine*: *uncertainty_range*) := **ACR-**object: *measurement_uncertainty* → **AMR-**object: *uncertainty_range*;
- AP_{16} (*determine*: *software_function*) := **ANIV-**objects: *S_filter* × *F_operator* × *L_filter* → **AME-**object: *software_functions*;
- AP_{17} (*determine*: *max_sampling_distance* × *max_sphere_radius*) := **APV-**object: *surface_type* × **ANIV-**object: *S_filter* → **AEV-**objects: *max_sampling_distance* × *max_sphere_radius* (It is mapped from AP_4);
- AP_{18} (*determine*: *max_sampling_distance* × *max_lateral_period_limit*) := **APV-**object: *surface_type* × **ANIV-**object: *S_filter* → **AEV-**objects: *max_sampling_distance* × *max_lateral_period_limit* (It is mapped from AP_5);
- AP_{19} (*determine*: *evaluation_area* × *sampling_length*) := **ANIV-**objects: *F_operator* × *L_filter* → **AEV-**objects: *evaluation_area* × *sampling_length* (It is mapped from AP_6);
- AP_{20} (*determine*: *S_filter* × *L_filter*) := **AFV-**object: *S-L_surface* → **ANIV-**objects: *S_filter* × *L_filter* (It is mapped from AP_7);
- AP_{21} (*determine*: *S_filter* × *F_operator*) = **AFV-**object: *S-F_surface* → **ANIV-**objects: *S_filter* × *F_operator* (It is mapped from AP_8).

As shown in figure 16, there are nine categories in the category model in verification, only five of themare mapped form the source categories (**ATD**, **AP**, **AE**, **AF** and **ANI**). Most of objects in three categories (**AME**, **ACR** and **AMR**) can be inferred by pullbacks from the objects of the five categories. However, some of the inferred results are for guides/suggestions only. The final decision is depend on the metrologists. For example, if the pullback AP_{12} infer the *instrument_type* will be stylus(contact stylus scanning), focus (focus variation microscopy) or SEM(scanning electron microscopy), it is the metrologists to decide which kind of instrument will be applied in the actual verification operators.

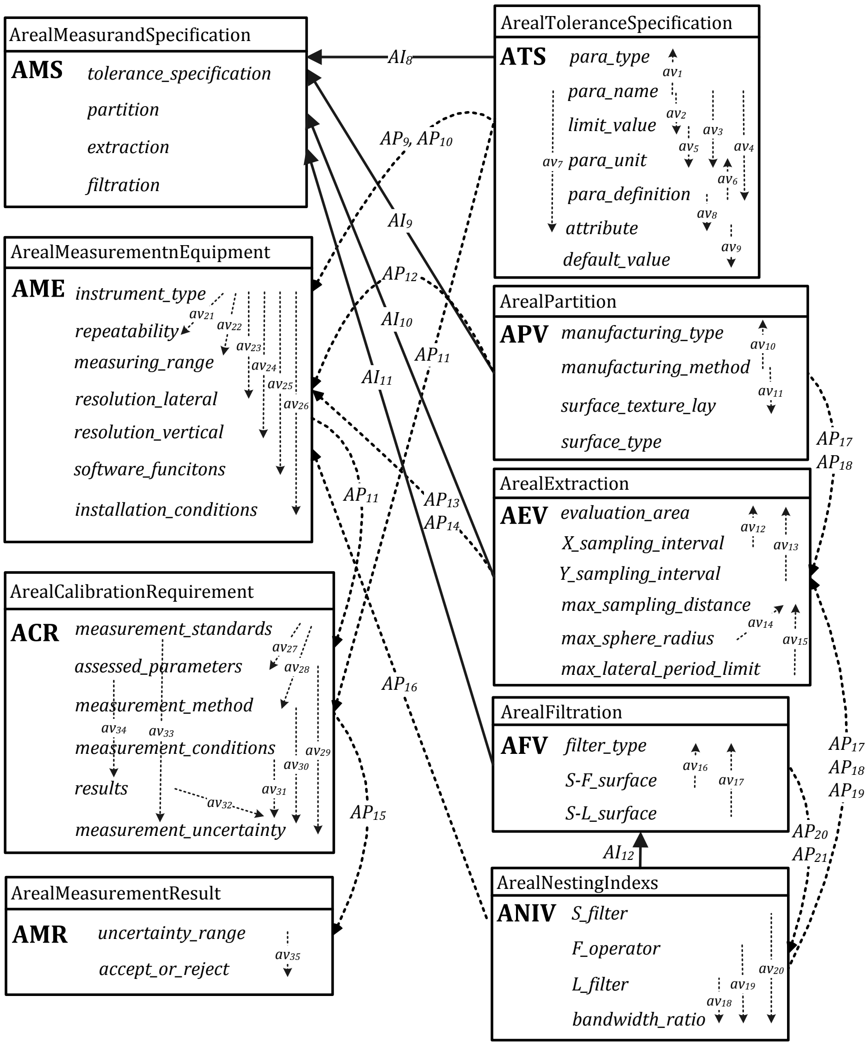


Figure 16. The category model for areal surface texture verification (high-level abstract diagram)

Figure 17 gives an example of pullback structure AP_{11} - the deduction of **ACR**-objects *measurement_standards* and *assessed_parameters*. The product of **ATS**-object *para_type* and **AME**-

object *instrument_type* determines **ACR**-objects *measurement_standards* and *assessed_parameters*. In the pullback structure, the objects *para_type* and *instrument_type* from the product of categories **ATS** and **AME** constitute a subcategory **SATM**. Since $\pi_1 p_{11} \circ \lambda_1 p_{11} = \pi_2 p_{11} \circ \lambda_2 p_{11}$, (**SATM×ACR**, $\pi_1 p_{11}$, $\pi_2 p_{11}$) is the pullback of (AP_{11} (…), $\lambda_1 p_{11}$, $\lambda_2 p_{11}$). The pullback structure AP_{11} means that the specified areal surface texture parameter type and related features of measurement instrument determine the type of measurement standard and related assessed parameters in calibration process. As data examples of AP_{11} are shown in Table 4, for an areal height parameter, if the calibration applies to measuring instrument that has a limited vertical measuring range and no arcuate motion correction, the suggested standards will be types of ER2, ER3, CG1 or CG2 (see ISO 25178-701:2010 [16]). For standard type of ER2, the assessed parameters are distance l_1 and l_2 between the grooves; for type of ER3, it is diameters D_f along the X- axis and the Y-axis. When the specified parameter is height or function type, if the calibration applies to measuring instrument having a large vertical measuring range and an arcuate motion correction, the suggested standard will be type of ES and related assessed parameters are diameters D_i along X-axis and Y-axis.

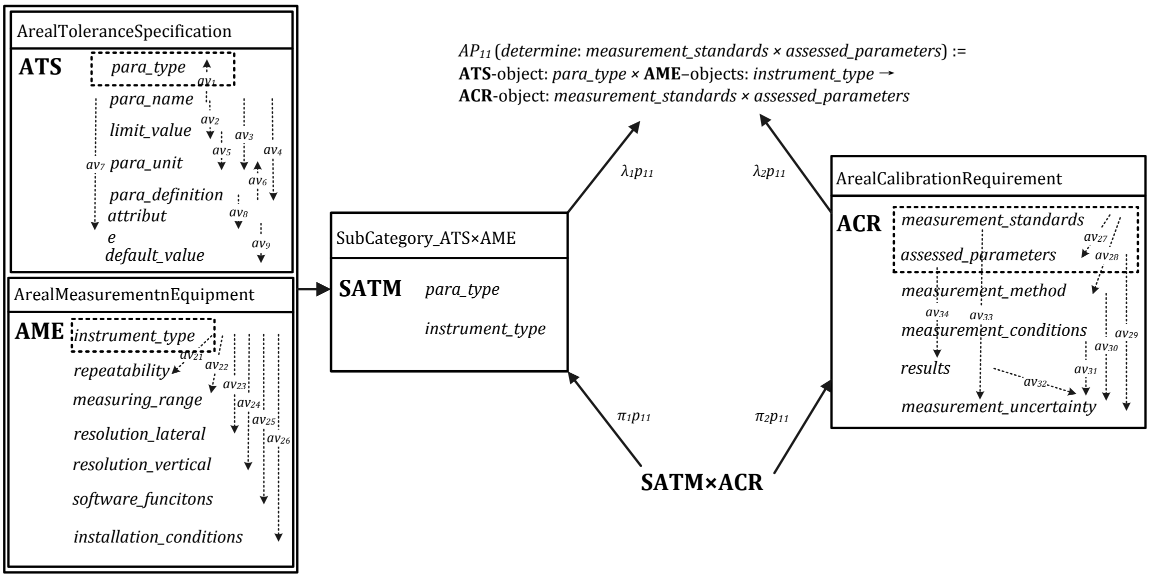


Figure 17. An example of pullback AP_{11} – the determination process of **ACR**-objects *measurement_standards* and *assessed_parameters*

ATS	AME	ACR	
para_type	*instrument_type*	*measurement_standards*	*assessed_parameters*
Height parameters	Instruments have a limited vertical measuring range and no arcuate motion correction	Standard ER2, ER3, CG1 or CG2	For ER2: distance l_1 and l_2 between the grooves For ER3: diameters D_f along the X-axis and the Y-axis
Height and function parameters	Instruments have a large vertical measuring range and an arcuate motion correction	Standard ES	Diameter D_i along X-axis and Y-axis
Spatial parameters	Instruments have a large measuring range and an arcuate motion correction	Standard ER2, ER3 or ES	Δ_{PER} (see ISO 25178-601 [11])

Table 4 Data examples of pullback AP_{11}

5. CONCLUSIONS

This paper utilizes category theory to model the diverse and sophisticated knowledge for specifications and verification in the field of areal surface texture. Categories and objects are applied to represent different knowledge structures; arrows and pullbacks are used to diagram diverse connection between objects; functors are utilized to reveal the structure-preserving mapping between categories in specification and verification. In particular, the pullbacks in this paper can be considered as a pullback inference mechanism, and most of the objects can be determined by the pullbacks.

The utilization of the category model enables the diagramming of sophisticated knowledge in areal surface texture regardless of details for structures or connections. As the development of areal surface texture standards are still in progress, much modification and updating is needed as well as publishing the areal surface texture standards. This diagramming modeling method makes it easier to update for programme designers. The knowledge model in this paper is the foundation for developing the areal surface texture design and measurement guide system for mechanical designers and metrologists.

ACKNOWLEDGEMENTS

The authors would like to thank Mr. Ding Hao and Mr. Francesco Ricci for their valuable suggestions, and thank Mr. David Brook for his kind help in this paper.

REFERENCES

[1] L. Blunt, K.J. Stout, Three-dimensional surface topography, Penton press, London, 2000.

[2] D.J. Whitehouse, The parameter rash – is there a cure? Wear. 83(1) (1982) 75-78.

[3] K.J. Stout, L. Blunt, W. Dong, et al, Development of method for the characterization of roughness in three dimensions, Butterworth Heinemann, London, 2002.

[4] L. Blunt, X. Jiang, Advanced techniques for assessment surface topography – Development of a basis for the 3D Surface Texture Standards "SURFSTAND", Kogan Page Science, www.kogenpagescience.com, 2003.

[5] ISO/FDIS 25178-2 Geometrical product specifications (GPS) – Surface texture: Areal – Part 2: Terms, definitions and surface texture parameters. 2010.

[6] ISO/TR 14638:1995 Geometrical product specification (GPS) – Masterplan. 1995.

[7] ISO/CD 25178-1 Geometrical product specifications (GPS) – Surface texture: Areal – Part 1: Indication of surface texture. 2009.

[8] P.J. Scott, Feature parameters, Wear. 266(2009) 548-551.

[9] ISO/DIS 25178-3 Geometrical product specifications (GPS) – Surface texture: Areal – Part 3: Specification operators. 2010.

[10] ISO 25178-6:2010 Geometrical product specifications (GPS) – Surface texture: Areal – Part 6: Classification of methods for measuring surface texture. 2010.

[11] ISO 25178-601:2010 Geometrical product specifications (GPS) – Surface texture: Areal – Part 601: Nominal characteristics of contact (stylus) instruments. 2010.

[12] ISO 25178-602:2010 Geometrical product specifications (GPS) – Surface texture: Areal – Part 602: Nominal characteristics of non-contact (confocal chromatic probe) instruments. 2010.

[13] ISO/DIS 25178-603 Geometrical product specifications (GPS) – Surface texture: Areal – Part 603: Nominal characteristics of non-contact (phase-shifting interferometric microscopy) instruments. 2010.

[14] ISO/DIS 25178-604 Geometrical product specifications (GPS) – Surface texture: Areal – Part 604: Nominal characteristics of non-contact (coherence scanning interferometry) instruments. 2010.

[15] ISO/CD 25178-605 Geometrical product specifications (GPS) – Surface texture: Areal – Part 605: Nominal characteristics of non-contact (point autofocus probe) instruments. 2009.

[16] ISO 25178-701:2010 Geometrical product specifications (GPS) – Surface texture: Areal – Part 701: Calibration and measurement standards for contact (stylus) instruments. 2010.

[17] ISO/WD 25178-702 Geometrical product specifications (GPS) – Surface texture: Areal – Part 702: Calibration and measurement standards for non-contact (confocal chromatic probe) instruments. 2010.

[18] ISO/CD 25178-70 ISO/CD 25178-70 Geometrical product specification (GPS) -- Surface texture: Areal -- Part 70: Physical measurement standards. 2010.

[19] ISO/DIS 25178-71 Geometrical product specifications (GPS) – Surface texture: Areal – Part 71: Software measurement standards. 2010.

[20] M. Barr, C. Wells, Category theory for computing science, Prentice Hall International, Hemel Hempstead, 1995.

[21] B.C. Pierce, Basic category theory for computer scientists, The MIT press, London, 1991.

[22] Y. Wang, A Knowledge-based Intelligent System for Surface Texture (VirtualSurf) in Department of Computing & Engineering, The University of Huddersfield, Huddersfield, 2008.
[23] Q. Qi, X. Jiang, X. Liu, et al, An unambiguous expression method of the surface texture, Measurement. 43 (2010) 1398-1403.
[24] W. Lu, X. Jiang, X. Liu, et al, Modeling the integration between specifications and verification for cylindricity based on category theory, Measurement Science and Technology. 21 (2010) 115107.
[25] ISO/TS 17450-1:2005 Geometrical product specifications (GPS) – General concepts – Part 1: Model for geometrical specification and verification. 2005.
[26] ISO 1302:2002 Geometrical product specifications (GPS) – Indication of surface texture in technical product documentation. 2002.